

X-670-71-406

PREPRINT

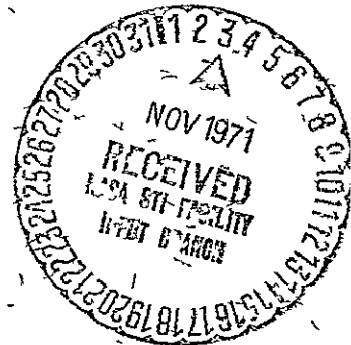
NASA TM X-65723

AN INTERACTIVE GRAPHICAL DATA SET MAINTENANCE PROGRAM

EDGAR M. GREVILLE
CLARENCE GROOVER

SEPTEMBER 1971

Reproduced by
NATIONAL TECHNICAL
INFORMATION SERVICE
Springfield, Va. 22151



GSFC

GODDARD SPACE FLIGHT CENTER

GREENBELT, MARYLAND

FACILITY FORM 602
171-37736
(ACCESSION NUMBER)
62
(PAGES)
TMX 65723
(NASA CR OR TMX OR AD NUMBER)

63 (THRU)
(CODE)
018
(CATEGORY)

AN INTERACTIVE GRAPHICAL DATA SET
MAINTENANCE PROGRAM

Prepared by

Edgar M. Greville*
and
Clarence Groover

Laboratory for Optical Astronomy

September 1971

GODDARD SPACE FLIGHT CENTER
Greenbelt, Maryland

*Computer Science Corporation

PRECEDING PAGE BLANK NOT FILMED

AN INTERACTIVE GRAPHICAL DATA SET
MAINTENANCE PROGRAM

Edgar M Greville
Clarence Groover

ABSTRACT

This document presents a description of the Interactive Graphical Data Set Maintenance Program, and discusses its application in the Goddard Experiment Package Operation Scheduling System for the OAO-B Satellite

CONTENTS

	<u>Page</u>
ABSTRACT	iii
INTRODUCTION	1
APPLICATION OF THE INTERACTIVE GRAPHICAL DATA SET MAINTENANCE PROGRAM	2
OPERATING PROCEDURES FOR PROGRAM DISPLAY	4
REQUIRED JOB CONTROL LANGUAGE CARD SETUP	10
PROGRAM PROCEDURE	11
SUBROUTINE GENERAL DESCRIPTION	12
DISPLY	12
GETCDS	13
CDIMGE	13
DSTBLE	14
TITLPG	14
LGTPEN	14
MODSET	15
MODIFY	15
APPENDIX I -- SAMPLES OF DISPLAY INFORMATION	
I-A TITLE PAGE	18
I-B DATA SET OPTION SELECT TABLE	19

CONTENTS (continued)

	<u>Page</u>
I-C CARD IMAGES WITH COLUMN LABELS	20
I-D CARD IMAGES WITHOUT COLUMN LABELS	21
APPENDIX II — SUBROUTINE FLOW CHARTS	
II-A SUBROUTINE DISPLY	23
II-B SUBROUTINE GETCDS	35
II-C SUBROUTINE CDIMGE	36
II-D SUBROUTINE DSTBLE	40
II-E SUBROUTINE TITLPG	41
II-F SUBROUTINE LGTPEN	42
II-G SUBROUTINE MODSET	43
II-H SUBROUTINE MODIFY	44
APPENDIX III — PROGRAM LISTINGS	
III-A SUBROUTINE DISPLY	47
III-B SUBROUTINE GETCDS	51
III-C SUBROUTINE CDIMGE	53
III-D SUBROUTINE DSTBLE	54
III-E SUBROUTINE TITLPG	55
III-F SUBROUTINE LGTPEN	56
III-G SUBROUTINE MODSET	57
III-H SUBROUTINE MODIFY	58

AN INTERACTIVE GRAPHICAL DATA SET MAINTENANCE PROGRAM

INTRODUCTION

The program described in this document was developed to provide an efficient means of maintaining several data sets in a production environment with a minimized likelihood of error. The program makes use of the IBM 2250 Graphical Display Unit to display selected records from one of the input data sets. In a series of simple, straightforward operations the 2250 operator may read a data set, modify the data, check his modifications and then write the modified data set back onto the device from which it was read. When making these modifications, the operator uses a function keyboard (a panel of buttons, each having a specific program assigned function), a light pen used to select items of data from the CRT screen, and a typewriter keyboard for entering characters into the data records which are displayed.

The data being modified is observable on the CRT before, during and after the modification process. Consequently the probability of committing errors of a typographical nature is greatly reduced. Also the data set need not be "written out" until the operator is convinced that all changes have been made satisfactorily. Important too is the fact that, since sixteen records are displayed simultaneously, the operator may view the record of interest in the context of the data stream.

The program may be treated as a subroutine and run in conjunction with processing programs under the control of a driver or executive program, or it may be run "off-line," as an independent utility program.

The program, as described here, is in subroutine form, with the symbolic name DISPLY. It is called by a driver routine which controlled the OAO-B Operations Scheduling System, for the purpose of providing means for modifying both input and output data sets under program control.

While the Graphical Data Set Maintenance Program was developed as an integral part of the OAO-B Operations Scheduling System, it is an independent self-contained program segment which could easily be adapted to a variety of needs involving graphical data set maintenance.

APPLICATION OF THE INTERACTIVE GRAPHICAL DATA SET MAINTEN- ANCE PROGRAM

To best understand the function, application and versatility of the Graphical Data Set Maintenance Program, a description of the application for which it was originally designed may be useful

During development of the OAO-B Operations Scheduling Software, it became obvious that contingencies might arise which would be difficult to handle in spite of the flexibility of input formats and the various program modes of operation. That is, the principles used to generate an experimentation schedule and command sequence, while perfectly adequate under normal operating conditions, would probably not be sufficient to cope with special environmental or spacecraft problems

It was thus determined that if the input data sets, stored on 2314 disk units, as well as the output data sets, also on disk, could be monitored and/or modified at strategic points during program execution, then the flexibility of the entire system could be tremendously enhanced

The scheduling system without the display package can be viewed as three modules, a read or input module, the calculation module, and the output module

Looking at Figure 1, we may see the logical flow of the scheduling system after inclusion of the Graphical Data Set Maintenance Program. Note that it is treated as a separate module. This is quite true in practice since the interface between the display module and other program segments consists only of the I/O devices and the attributes of the data format

From Figure 1, it can be seen that the display module also provides for branching to different program segments by executing FORTRAN optional return statements of the form RETURN 1

For example, after executing the read module, the driver program calls subroutine DISPLY, the symbolic name of the display module. This allows the program operator to look at the input data sets exactly as read and if necessary, to modify the data and then go back and re-execute the read module

Once it has been determined that the inputs are correct, a normal return results in the execution of the computations module. After the computations are finished, the display module is again entered and the operator may examine the output data sets and make any modification to the outputs he desires before returning for execution of the output module which writes the schedule information on permanent data sets. If after looking at the program outputs the operator

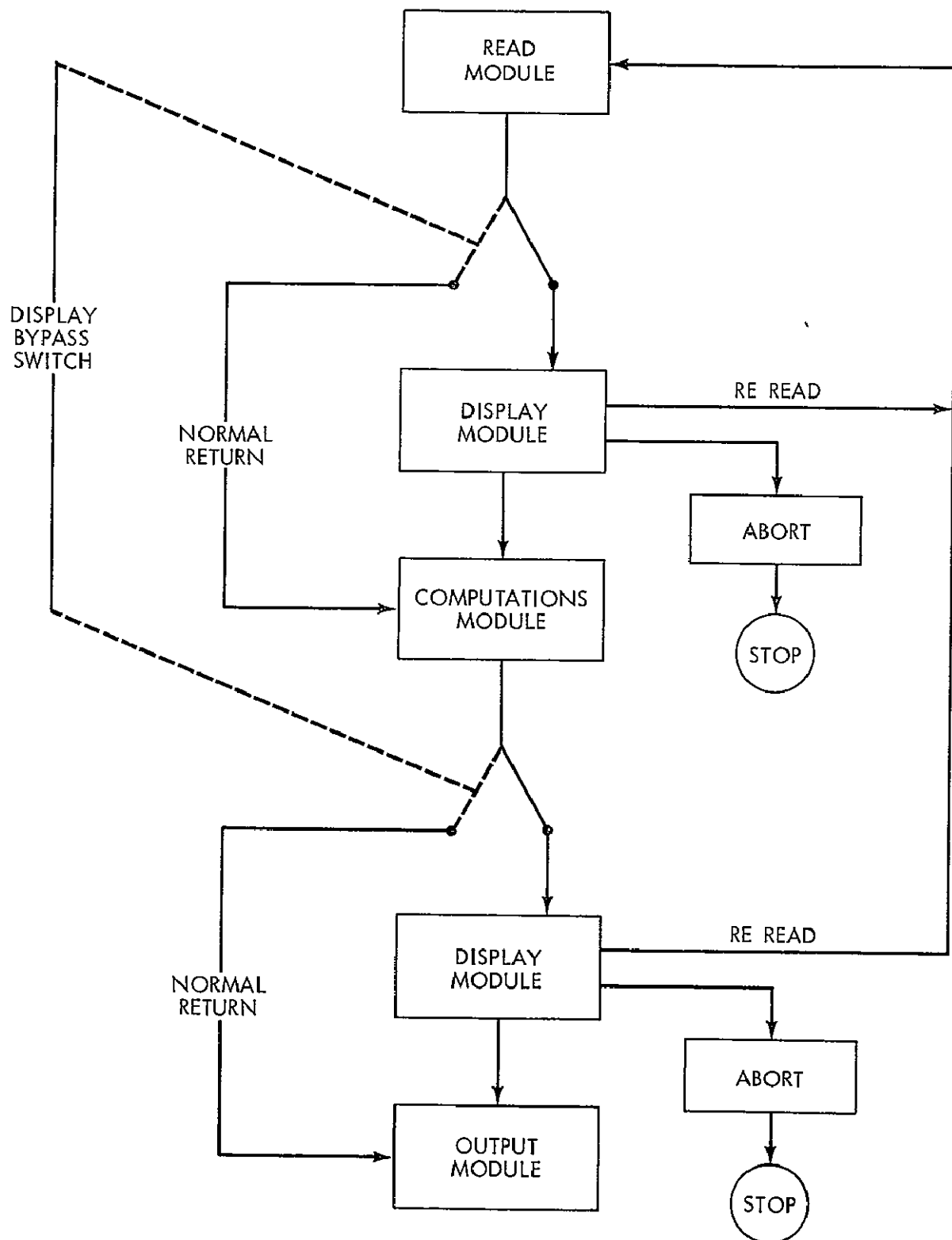


Figure 1

feels that the program should be re-run, he may modify the inputs and, utilizing the optional return feature, cause the data to be re-read

The program may be re-executed in this fashion as many times as the operator desires within the limits of the time estimate

Thus, with this type of structure the Graphical Data Set Maintenance Program may be used to bring a person "into the loop" of a convergent-type process as well as for routine editing and utility purposes

OPERATING PROCEDURES FOR PROGRAM DISPLAY

- 1 Depress Function Key #4
 - Erases title page
 - Displays option select table (see Appendix I-B)
- 2 Point Light Pen at Desired Data Set
 - Erases option select table
 - Displays selected data set and corresponding unit number
 - Displays modification options (see Appendix I-C)
- 3 Depress Function Key #1
 - Reads data set from disc
 - Sounds alarm when read is complete
- 4 Depress Function Key #2
 - Displays 16 card images (see Appendix I-D)
- 5 Depress Function Key #5
 - Erases 16 card images
- 6 Repeat Steps 4 and 5 for number of cards desired to be scanned

To Modify a Card Image

- 7 Point light at option desired to perform corresponding operation
 - Option selected is displayed on lower right of screen
 - Modification options are erased
 - Curser appears in Column 1 of card image under consideration
 - Card sequence number appears at the end of card image under consideration (see Appendix I-D)
 - Lights on function key board are illuminated in the configuration of the number 2 indicating the 2nd attention level
- 8 Point light pen at character desired to be edited
 - Curser appears underneath character (see Appendix I-D)
- 9 Depress alternate code and end key
 - Transfers image into storage
 - Lights in key board are illuminated in the configuration of the number 1 indicating that you have returned to the first attention level
 - Updates the data set
 - Erases selected option
 - Erases the 16 card images
 - Erases card image sequence number
 - Displays modification options
 - Redisplays images including changes
- 10 Repeat steps 6-8 for number of card images desired to be modified
- 11 Depress Function Key #5
 - Erases 16 card images

12 Depress Function Key #14

- Writes output data set on disc
- Alarm sounds when write is complete

13 Depress Function Key #3

- Erases modification options
- Erases D set name and corresponding unit number
- Displays option select table (see Appendix I-B)

14 Repeat steps 4-13 for desired number of data sets

Attention Level Number 1

Function Key #7 - Omits Column Labels

Function Key #6 - Displays Column Labels (see Appendix I-C)

Function Key #8 - Back page 8 Card Images

Function Key #9 - Displays Entry Point & Card Image Isolation Information

Function Key #10 - Redisplays 16 card images that are presently displayed

Function Key #11 - Select card image independent of light pen (see Appendix I-D)

Function Key #12 - Select Data Set Name independent of light pen (see Appendix I-C)

Function Key #13 - Selection Modification Function independent of light pen

Function Key #16 - Back page 16 card images

End Key - Transfers Entry point and Backup mode Data into storage

Function Key #29 - Terminates Job

Attention Level Number 2

Function Key #19 will take you back to Attention Level #1

NOTE

Do Not Select Dummy From Option Select Table

To Start At Any Card Image In The Data Set

A Depress Function #9

- EP0000 and CI0000 appears at lower left of screen
- Curser appears under the most significant digit of EP

B Enter card image sequence number

C Depress alternate code and END key

- Transfer number into code
- Erases EP0000 and CI0000

D Depress Function Key #2

- Displays 16 card images beginning at the one specified

To select Data Set from Option Select Table independent of light pen

I Depress Function Key #9

- CI0000 appears with curser in most significant digit

II Type in Value from Table 1 corresponding to desired Data Set

III Depress Altncode end key

- CI0000 is erased

IV Depress Function Key #12

- Same as Step #2

Table 1

DATAIN -- 3400	TARGETS -- 1800
ORBELM -- 3200	GEPCMD -- 1600
EXPMOD -- 3000	GEP SHD -- 1400
CALIBR -- 2800	OPRLOG -- 1200
NETCOM -- 2600	OBSLOG -- 1000
MODELS -- 2400	DUMMY -- 800
CDINPT -- 2200	DUMMY -- 600
PASPOS -- 2000	DUMMY -- 400

To modify Card Image independent of light pen

- a Depress Function Key #9
 - CI0000 appears with cursor
- b Type in value Table 2 corresponding to option
- c Depress Alternate Code End Key
 - CI0000 is erased
- d Depress Function Key #13
- e Repeat Step a-c substituting values from Table 3
- f Depress Function Key #12
 - Same as Step #7

Core requirements will be of the order of 120K depending on the size and number of I/O buffers

CPU and I/O time requirements should not exceed one-half minute each for normal execution

Response time is normally instantaneous unless the system is heavily loaded in which case interrupts may be somewhat slow

Table 2

D-3610
I-3785
E-3910

Table 3

IMAGE1	--	3810
IMAGE2	--	3510
IMAGE3	--	3310
IMAGE4	--	3110
IMAGE5	--	2810
IMAGE6	--	2610
IMAGE7	--	2410
IMAGE8	--	2110
IMAGE9	--	1910
IMAGE10	--	1610
IMAGE11	--	1310
IMAGE12	--	1110
IMAGE13	--	900
IMAGE14	--	610
IMAGE15	--	410
IMAGE16	--	200

REQUIRED JOB CONTROL LANGUAGE CARD SETUP

//GD.FT20F001	DD DSNNAME=K3.T2DKW.DSORBELM,DISP=OLD,	*
//	UNIT=2314,VOL=SER=G2GEP1	
//GD.FT21F001	DD DSNNAME=K3.T2DKW.DSEXPMOD,DISP=OLD,	*
//	UNIT=2314,VOL=SER=G2GEP1	
//GD.FT22F001	DD DSNNAME=K3.T2DKW.DSEXPCL,DISP=OLD,	*
//	UNIT=2314,VOL=SER=G2GEP1	
//GD.FT23F001	DD DSNNAME=K3.T2DKW.DSNETCON,DISP=OLD,	*
//	UNIT=2314,VOL=SER=G2GEP1	
//GD.FT24F001	DD DSNNAME=K3.T2DKW.DSMODFLS,DISP=OLD,	*
//	UNIT=2314,VOL=SER=G2GEP1	
//GD.FT25F001	DD DSNNAME=K3.T2DKW.DSCDINPT,DISP=OLD,	*
//	UNIT=2314,VOL=SER=G2GEP1	
//GD.FT26F001	DD DSNNAME=K3.T2DKW.DSPASPOS,DISP=OLD,	*
//	UNIT=2314,VOL=SER=G2GEP1	
//GD.FT27F001	DD DSNNAME=K3.T2DKW.DSTARCTS,DISP=OLD,	*
//	UNIT=2314,VOL=SER=G2GEP1	
//GD.FT28F001	DD DSNNAME=K3.T2DKW.DSGEPCMD,DISP=OLD,	*
//	UNIT=2314,VOL=SER=G2GEP1	
//GD.FT34F001	DD DSNNAME=K3.T2DKW.DSOPRLOG,DISP=OLD,	
//	VOL=SER=G2GEP1,UNIT=2314	
//GD.FT35F001	DD DSNNAME=K3.T2DKW.DSGEPSHD,DISP=OLD,	1
//	UNIT=2314,VOL=SER=G2GEP1	
//GD.FT36F001	DD DSNNAME=K3.OPRSCR,DISP=(NEW,DELETE),UNIT=SYSOUT,	1
//	DCB=(RECFM=FB,LRECL=80,BLKSIZE=3200),SPACE=(80,6000)	
//GD.FT37F001	DD DSNNAME=K3.T2DKW.DSOHSLG,DISP=OLD,	1
//	UNIT=2314,VOL=SER=G2GEP1	
//GD.FT49F001	DD UNIT=012	
//GD.FT15F001	DD DSNNAME=K3.T2DKW.DSEXPMOD,DISP=OLD,	1
//	UNIT=2314,VOL=SER=G2GEP1	
//	DD DSNNAME=K3.T2DKW.DSEXPCL,DISP=OLD,	X
//	UNIT=2314,VOL=SER=G2GEP1	
//	DD DSNNAME=K3.T2DKW.DSMODELS,DISP=OLD,	X
//	UNIT=2314,VOL=SER=G2GEP1	
//	DD DSNNAME=K3.T2DKW.DSGEPCMD,DISP=OLD,	1
//	UNIT=2314,VOL=SER=G2GEP1	
//	DD DSNNAME=K3.T2DKW.DSTARCTS,DISP=OLD,	X
//	UNIT=2314,VOL=SER=G2GEP1	
//	DD DSNNAME=K3.T2DKW.DSNETCON,DISP=OLD,	X
//	UNIT=2314,VOL=SER=G2GEP1	
//	DD DSNNAME=K3.T2DKW.DSORBELM,DISP=OLD,	1
//	UNIT=2314,VOL=SER=G2GEP1	
//	DD DSNNAME=K3.T2DKW.DSCDINPT,DISP=OLD,	
//	UNIT=2314,VOL=SER=G2GEP1	

6043 CAPDS-

PROGRAM PROCEDURE

The Interactive Graphical Data Set Maintenance Program makes use of two attention levels in accessing data sets from the card reader and disc storage, and allows the IBM 2250 operator to monitor or make changes to the data and then place the data sets back onto disc. Data sets are read, displayed and written onto disc at Attention Level 1. Data set records are modified at Attention Level 2.

The initialization subroutines are called which readies The Graphic Subroutine Package, The IBM 2250 CRT Display Unit and the Graphical Data Sets. The program then creates Attention Level 1, calls subroutines MODSET and TITLPG displaying a Title Page. (See Title Page display in Appendix I-A.) The program is then placed in a Wait status.

Upon depression of function key #4, the Title Page is erased. Subroutine DSTBLE is called displaying The Data Set Option Select Table containing the menu of data sets selectable at the choice of the operator. (See display of Option Select Table in Appendix I-B.) The operator then makes his selection by touching the data set name in The Option Select Table with the light pen. This erases the Option Select Table. The program calls subroutine LGTPEN which sets the unit number and displays the selected data set and unit number which it resides at the bottom of the screen. (Appendix I-C shows that data set TARGTS was selected and that it resides on unit 27.)

When function key #1 is depressed, the data set is read. The alarm is sounded when read is complete. Upon depression of function key #2, the program calls subroutine GETCDS which determines the starting point in the data set buffer and displays 16 card images. The modification options are also displayed. If column labels are desired, this is accomplished by depressing function key #6. Function key #7 will omit them.

The operator may scan the data set in a forward direction by depressing function key #5 which erases screen and function key #2 which displays the next 16 card images. Back paging of 8 or 16 card images can be accomplished by depressing function keys 8 and 16, respectively.

When modification to a card image is desired, the operator may select the image by touching the corresponding modification option with the light pen. (Appendix I-C shows the modification options beginning at the upper right and ending at the lower right of the screen. D, I and E means Delete, Insert and Edit, respectively.) The program will then erase the modifications options and call subroutine CDIMGE. Subroutine CDIMGE will move program control to

Attention Level Number 2, insert curser underneath the first character of the selected card image CDIMAGE also displays the data set buffer count for the selected card image and the modification operation which is to be performed (Appendix I-D shows the curser and card count of 7 displayed in the selected card image The modification operation EDIT is displayed on the lower right of the screen) If for some reason the operator does not wish to change the selected card image, program control is returned to Attention Level Number 1 by depressing function key #19 At this point the modification options are redisplayed which allows the operator to select any card image and modification operation desired

When modification operation DELETE is selected the program moves program control to Attention Level Number 2 and waits Upon depression of the alternate code and end key from the typewriter keyboard, the screen is erased Subroutine MODIFY is called updating the data set buffer Then control is returned to Attention Level Number 1 and the 16 images are redisplayed reflecting the deleted card image In the case of modification operations EDIT and INSERT, the changes are typed in before depressing the alternate code and end key

When the operator is satisfied that a data set is correct, he can write it on-to disc by depressing function key #14 The alarm will sound when the write is complete The program can be readied to select the next data set by depressing function key #3 which erases the screen and redisplayes the Data Set Option Select Table When all data sets have been verified for accuracy, the display program is ended by depressing function key #29

The program and all subroutines are written in Fortran IV to be run on the IBM System 360/65, 75 and 91 computers

SUBROUTINE GENERAL DESCRIPTIONS

DISPLY

This subroutine is the driver for the Interactive Graphical Data Set Maintenance Program It initializes the IBM 2250 Unit, Graphic Subroutine Package and the Graphical Data Sets It creates Attention Level Number 1 and contains the read and write logic It calls the following application subroutines TITLPG, MODSET, GETCDS, DSTBLE, LGTPEN and CDIMAGE

Calling Sequence

CALL DISPLY (IERNUM, *,*)

IERNUM Normal Return

* Return 1

* Return 2

See Flow Chart in Appendix II-A

GETCDS

This subroutine determines the entry point and transfers 16 card images into active area. It calculates the card count and call subroutine INCORE to convert the count to character code. Displays the card images and set KZ = 1

Calling Sequence

CALL GETCDS (N)

N Card sequence number for entry point card image

See Flow Chart in Appendix II-B

CDIMGE

This subroutine determines whether to DELETE, INSERT or EDIT. It then creates Attention Level Number 2, inserts cursor underneath first character of selected card image, displays card count for selected card image, and modification selected modification option. Allows IBM 2250 operator to modify and transfer selected card image into core. Also call subroutine MODIFY to make proper disposition of card image in update area and return control to subroutine DISPLY.

Calling Sequence

CALL CDIMGE (INAME, KYONE, KYTWO, KKEY, KL, KMAX, N, KZ)

INAME Device Name

KYONE Key for 1st portion of selected card image

KYTWO Key for 2nd portion of selected card image

KKEY Key for displaying card count for selected card image

KL Card count for selected card image

KMAX Number of card images in data set

N Count for first card on display unit

KZ If KZ not equal 1 the program will erase screen and redisplay
these 16 card images reflecting the changes

See Flow Chart in Appendix II-C

DSTBLE

This subroutine generates the Data Set Option Select Table

Calling Sequence

CALL DSTBLE

See Flow Chart in Appendix II-D

TITLPG

This subroutine displays Title Page information, and generates modification
and the column labels

Calling Sequence

CALL TITLPG

See Flow Chart in Appendix II-E

LPTPEN

This subroutine selects data set, determines which unit the data set resides,
and rewinds the unit

Calling Sequence

CALL LGTPEN (IUNIT, NCORRT, KDSET)

IUNIT Unit number

NCORRT Y-coordinate for data set

KDSET Key for displaying selected data set and
unit number at bottom of screen

See Flow Chart in Appendix II-F

MODSET

This subroutine generates data set and option selected information

Calling Sequence

CALL MODSET

See Flow Chart in Appendix II-G

MODIFY

This subroutine performs the necessary function on the active area to
INSERT, DELETE or EDIT a card image

Calling Sequence

CALL MODIFY (KL, KMAX, N)

KL Count for selected card image

KMAX Number of card images contained in data set

N Count for 1st card image displayed on screen —

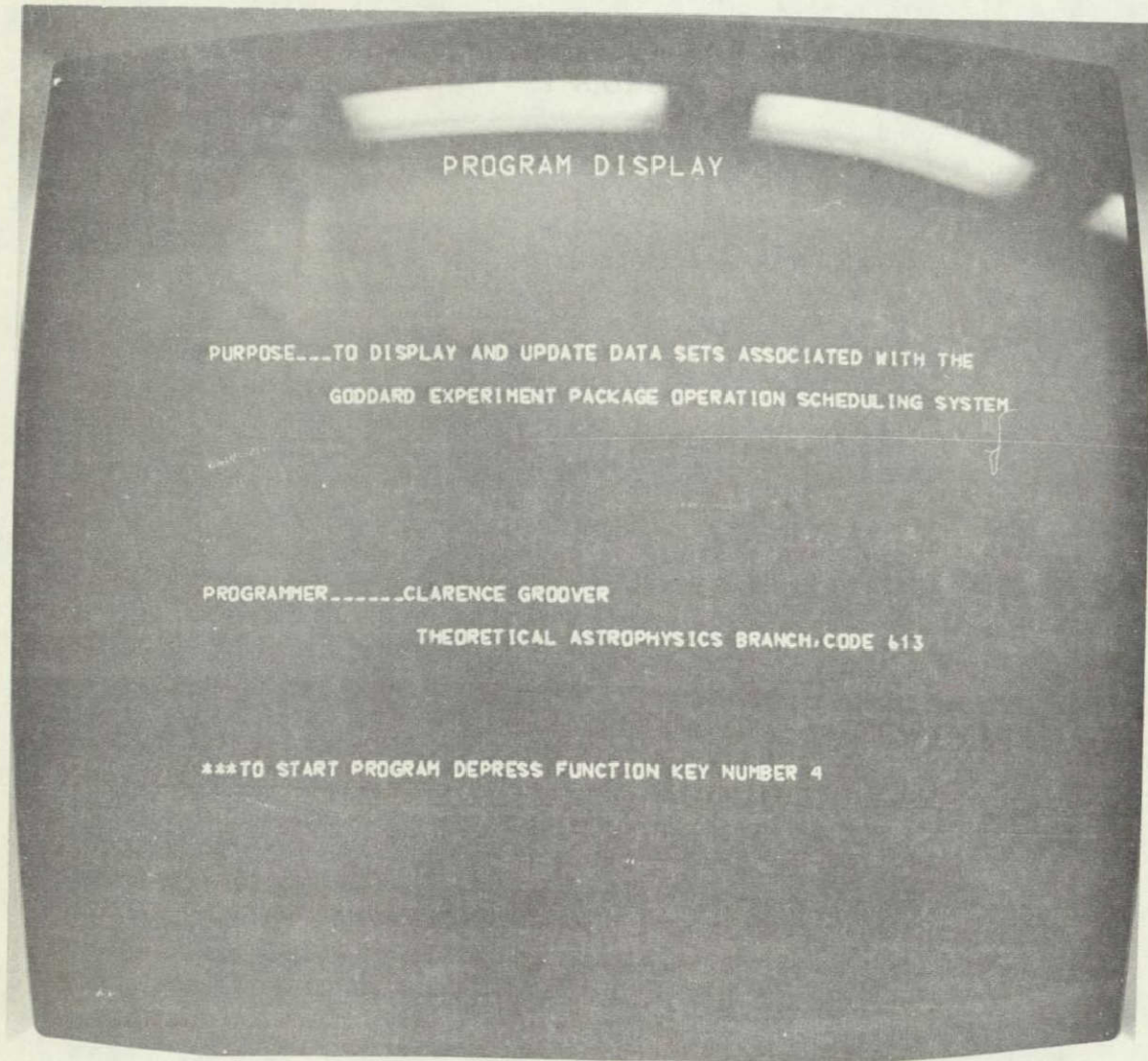
See Flow Chart in Appendix II-H

PRECEDING PAGE BLANK NOT FILMED

APPENDIX I

APPENDIX I-A

TITLE PAGE



APPENDIX I-B

DATA SET OPTION SELECT TABLE

DATA SET OPTION SELECT TABLE	
DATA SET NAME	UNIT NUMBER
DATAIN	15
ORBELM	20
EXPMOD	21
CALIBR	22
NETCON	23
MODELS	24
CDINPT	25
PASPOS	26
TARGTS	27
GEPCMD	28
GEPSHD	35
OPRLOG	36
OBSLOG	37
DUMMY	38
DUMMY	39
DUMMY	5

***POINT LIGHT PEN AT THE DESIRED DATA SET NAME

NOT REPRODUCIBLE

CARD IMAGES WITH COLUMN LABELS

DS_NAME: TARGTS UNIT#: 27

APPENDIX I-D

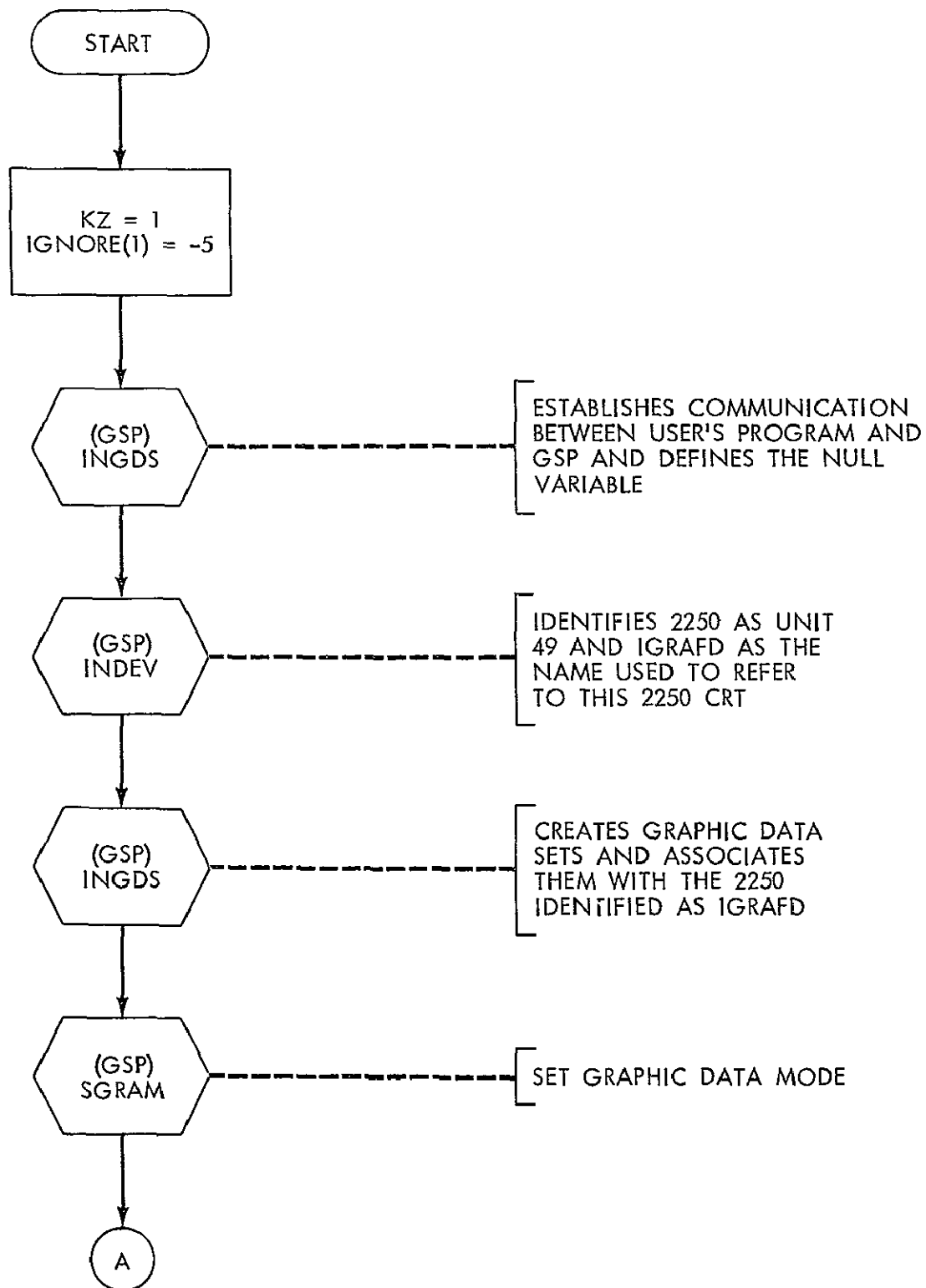
CARD IMAGES WITHOUT COLUMN LABELS

TARGET LIST											
S	HOLD	5	40.00	10	00.00					15	
S 034085		5	12.133-	8	15.48	B81	0.10	-0.03	0.0	8	0 A
BUA1	+										
S 053138		7	0.933-23	55.55	B31	3.10	-0.08		0.0	-302	0 A
BUA1	+										
S 058350		7	22.117-29	12.27	B51	2.40	-0.07		0.0	-302	0 A
BUA1	+										
S 036629		5	30.483-	4	35.77	B53	8.00	0.02	0.0	DARK -412	AN2 D
AKH*	+										
S 037043		5	32.950-	5	56.95	053	2.90	-0.23	0.0	-412	0 D
AKH*	+		7								
S 037058		5	33.083-	4	51.95	B53	7.40	-0.14	0.0	DARK -412	AN2 D
AKH*	+										
S 037129		5	33.583-	4	26.98	B53	7.00	-0.01	0.0	DARK -412	AN2 D
AKH*	+										
S 037807		5	38.600-	3	38.35	B83	8.60	-0.09	0.0	DARK -412	AN2 D
AKH*	+										
S 079158		9	10.583	43	25.75	B83	5.30	-0.14	0.0	DARK -412	AN2 D
AKH*	+										
S 022470		3	33.983-17	38.00	A03	5.30	-0.14		0.0	DARK -412	AN2 D
AKH*	+										
S 022920		3	38.183-	5	22.25	B83	5.50	-0.14	0.0	DARK -412	AN2 D
AKH*	+										
S 049333		6	44.883-20	57.60	B83	6.00	-0.12		0.0	DARK -412	AN2 D
AKH*	+										
S 061429		7	36.200	25	14.73	B83	4.60	-0.10	0.0	-412	0 D
AKH*	+										
S 069082		8	12.083-36	11.05	B83	6.00	-0.18		0.0	DARK -412	AN2 D
AKH*	+										
DS_NAME: TARGTS										UNIT#: 27	
										EDIT	

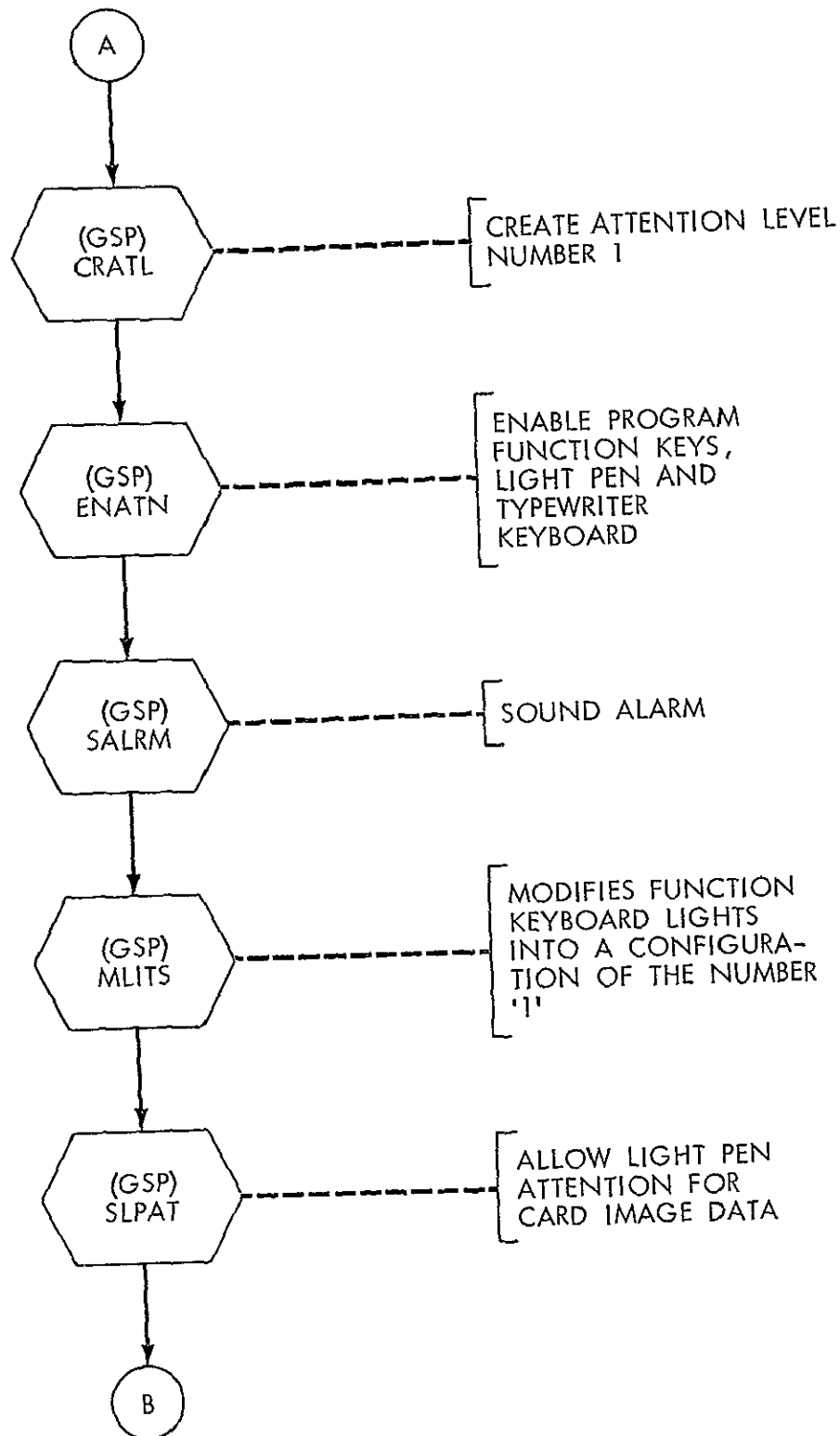
NOT REPRODUCIBLE

APPENDIX II-A

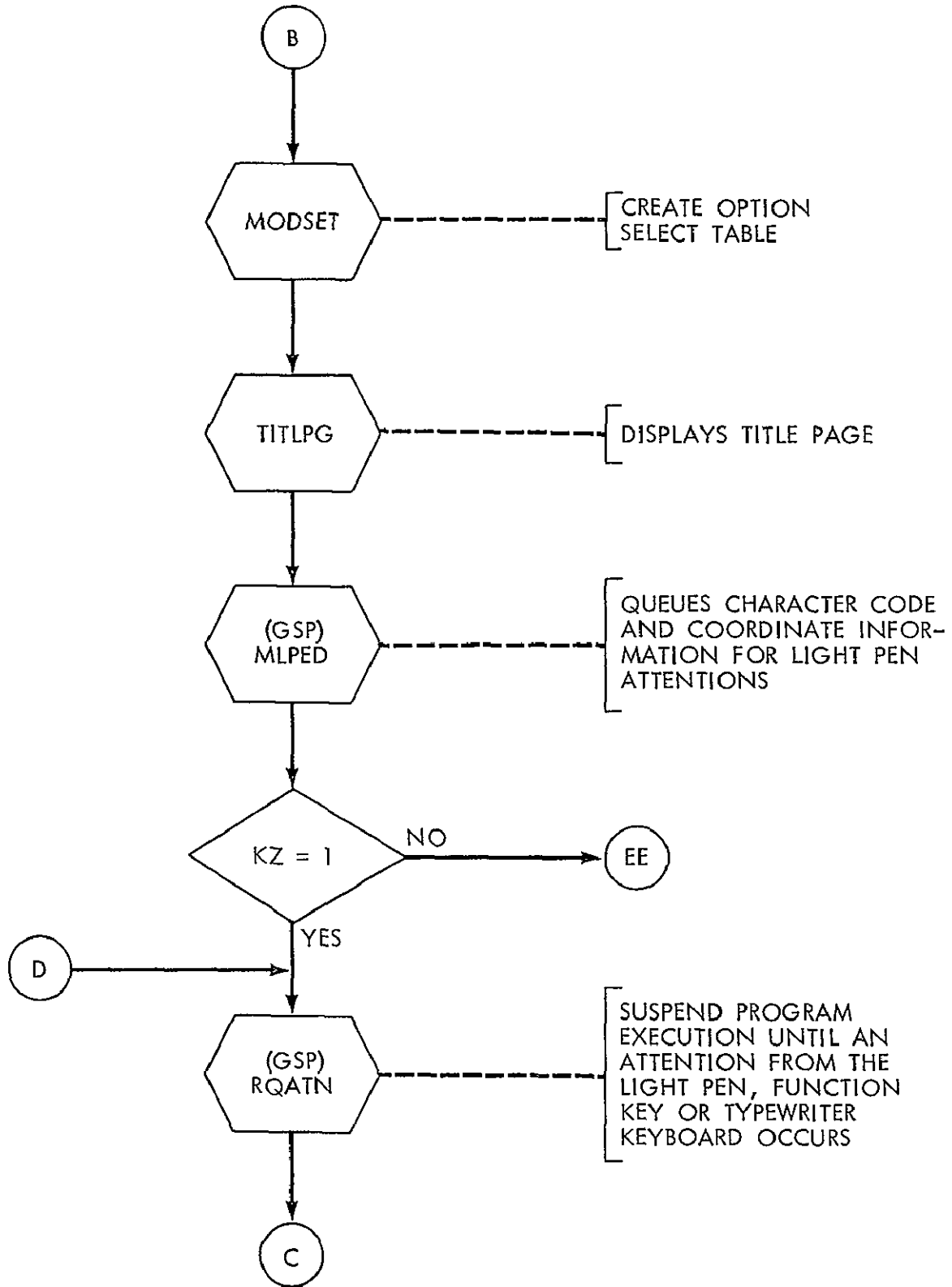
SUBROUTINE DISPLAY



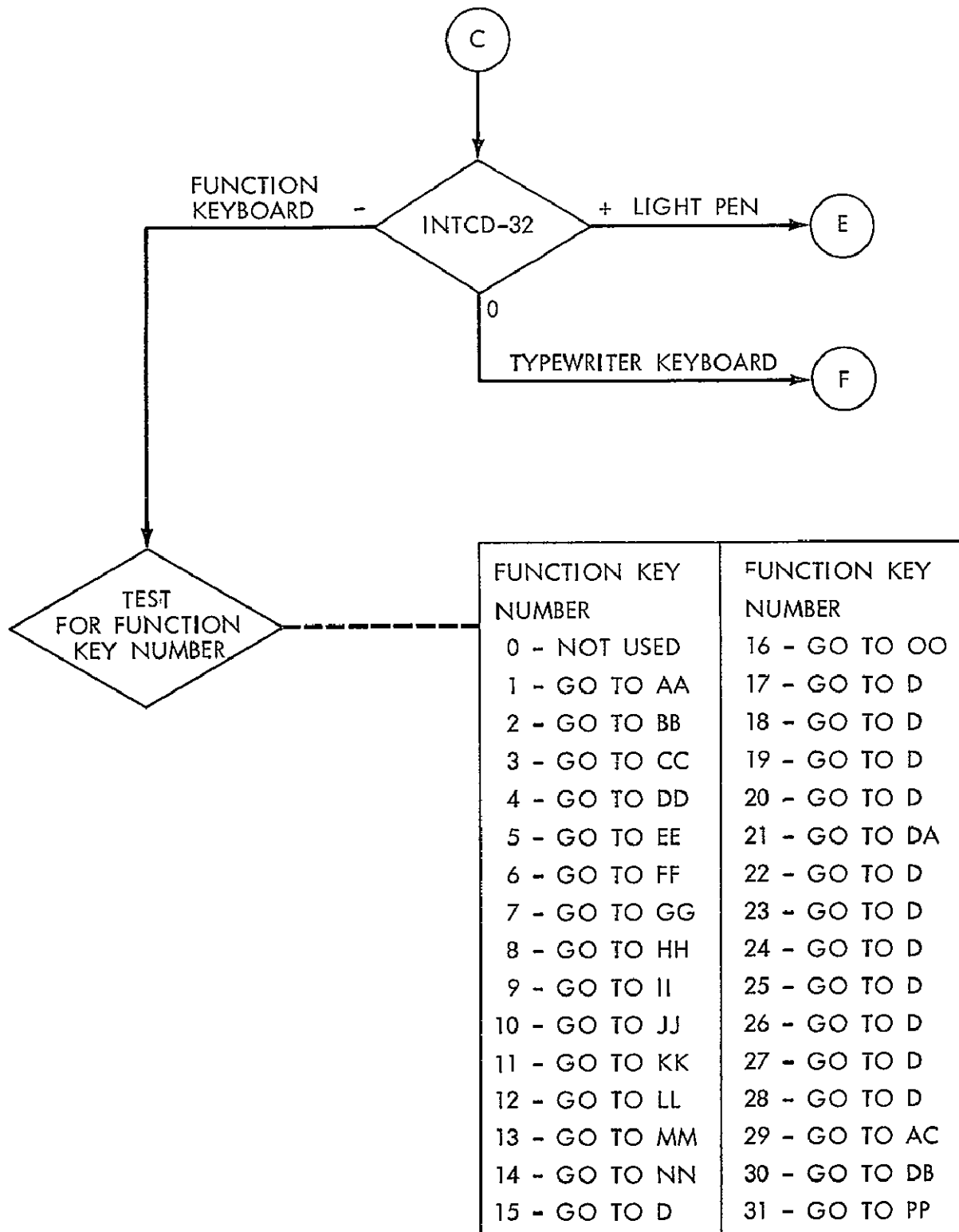
SUBROUTINE DISPLAY (continued)



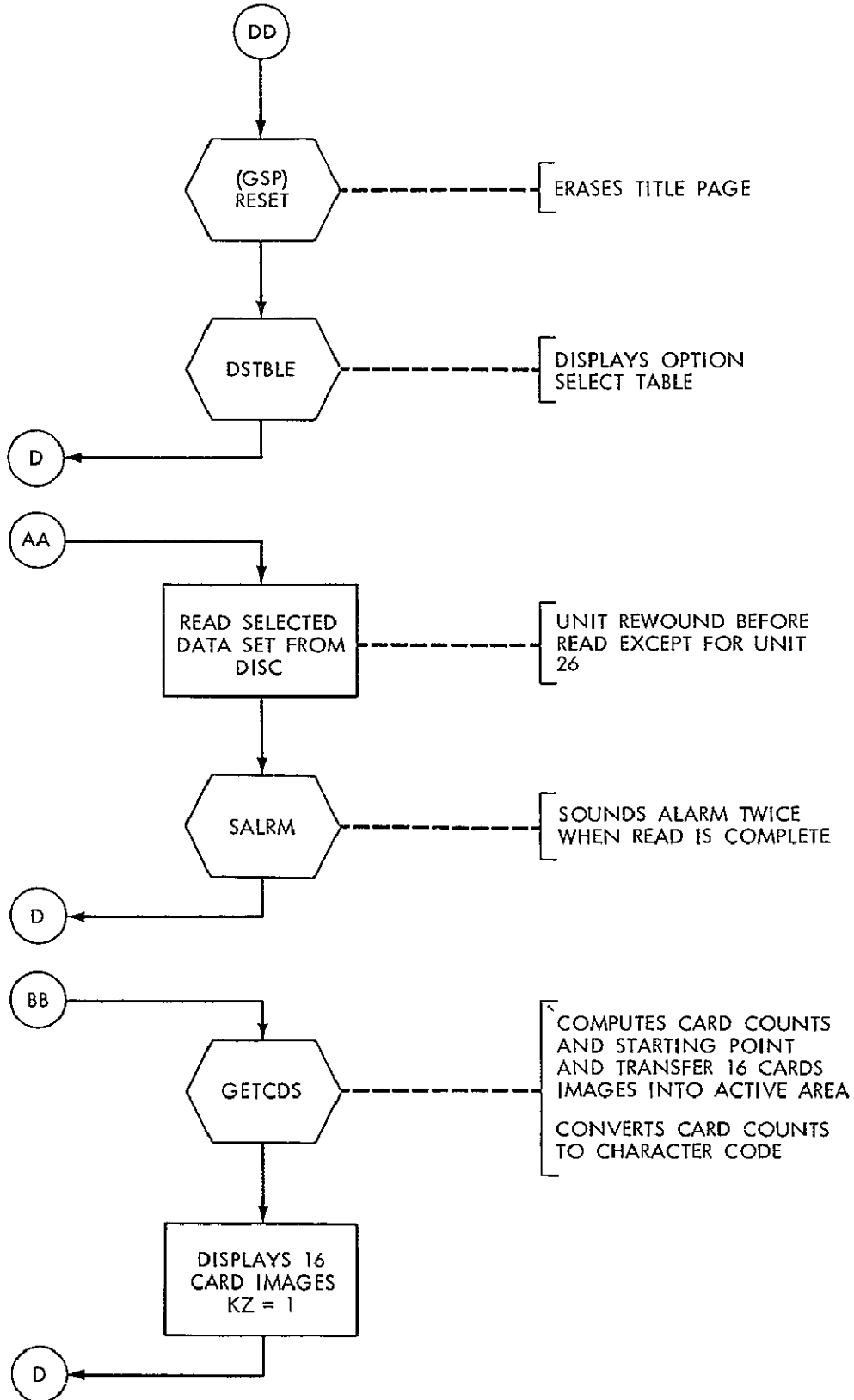
SUBROUTINE DISPLAY (continued)



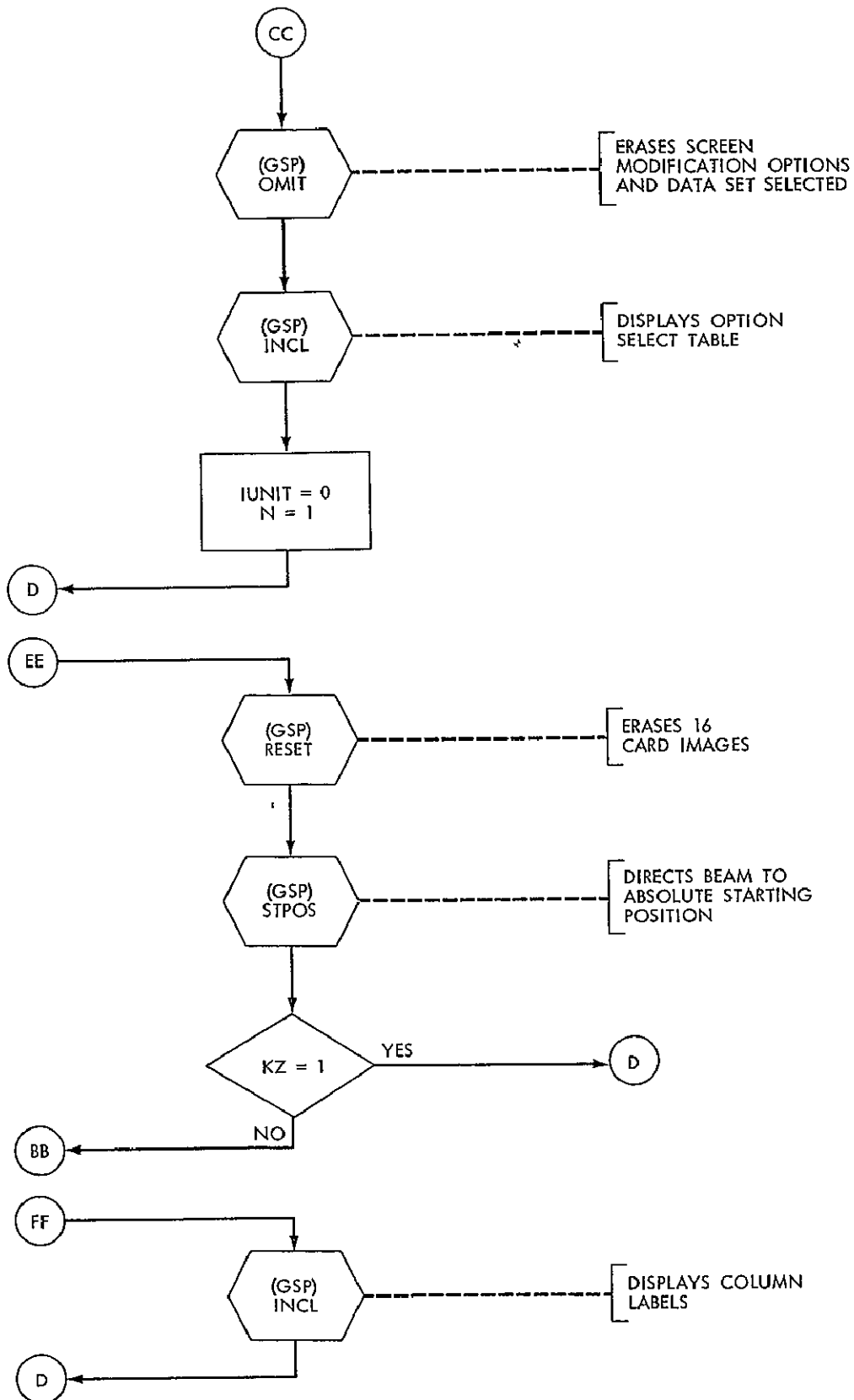
SUBROUTINE DISPLAY (continued)



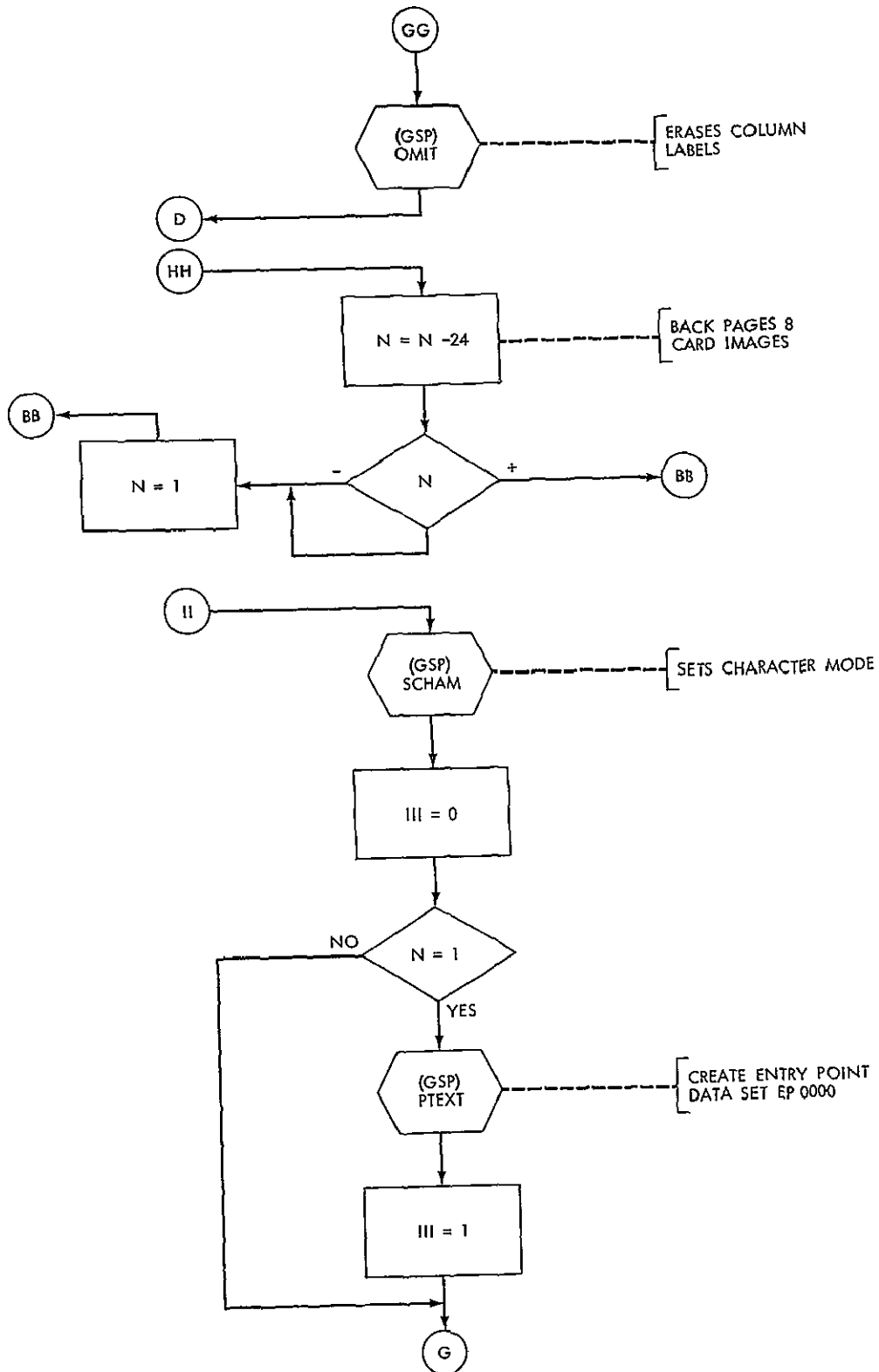
SUBROUTINE DISPLAY (continued)



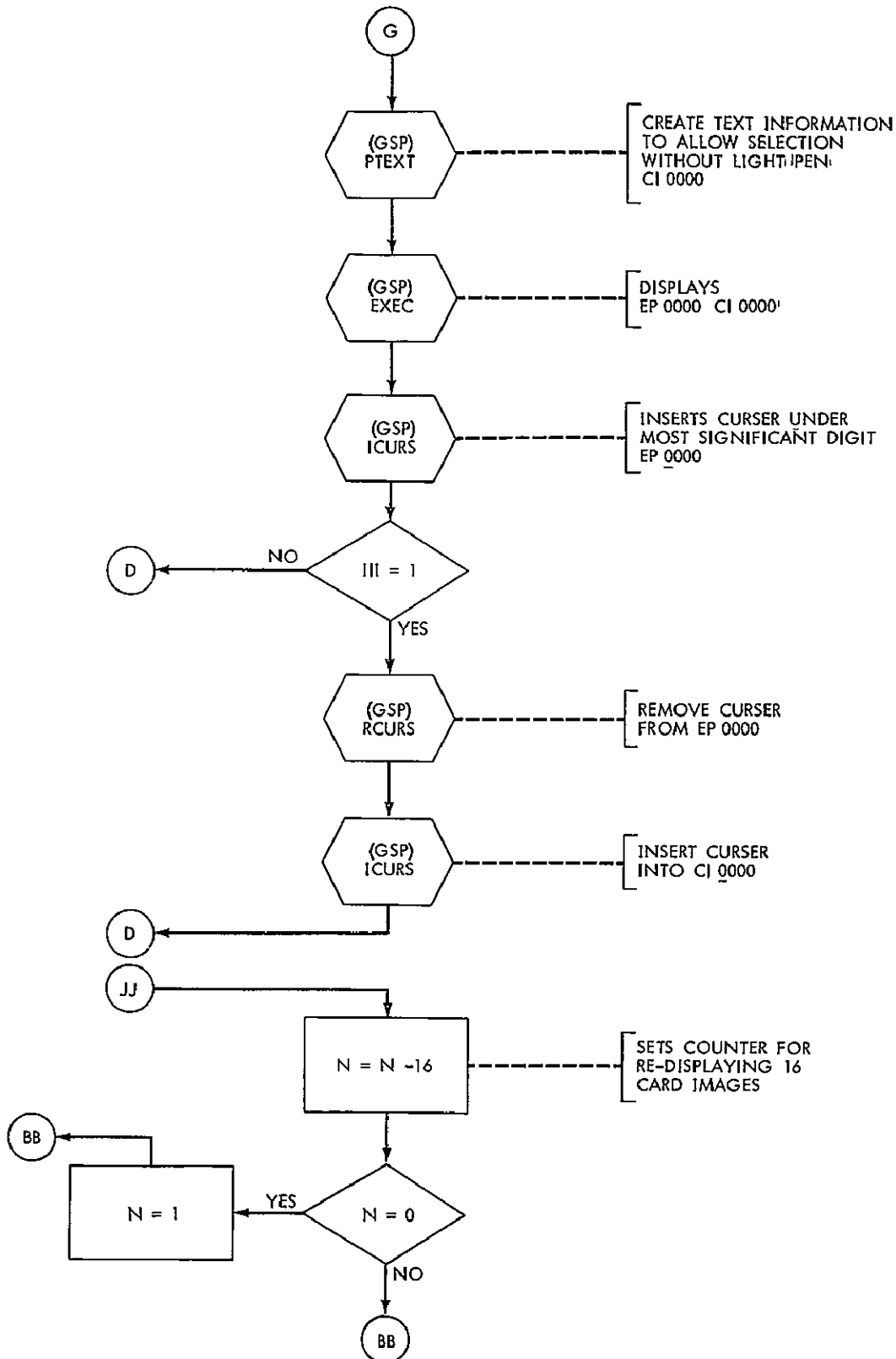
SUBROUTINE DISPLAY (continued)



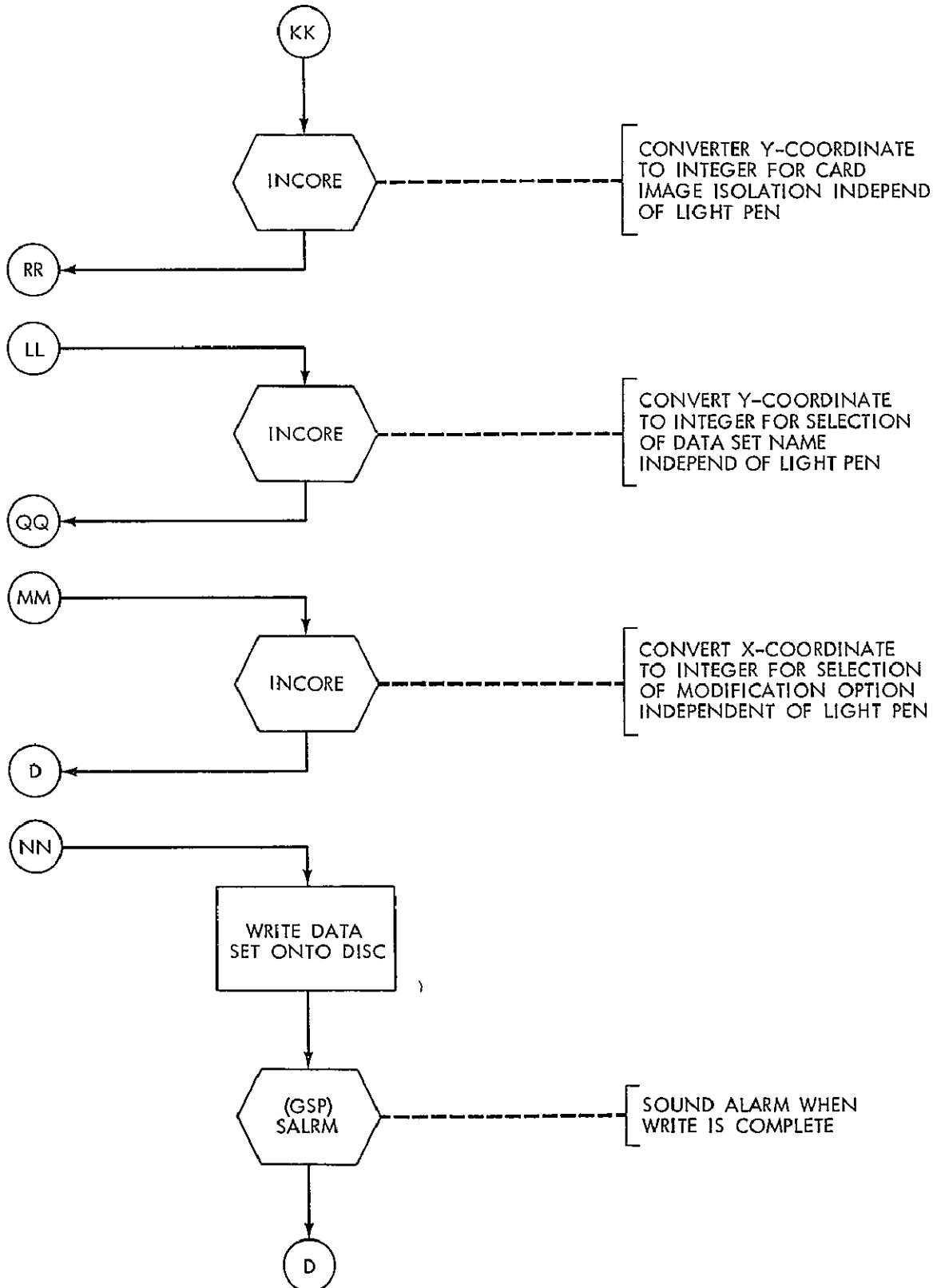
SUBROUTINE DISPLAY (continued)



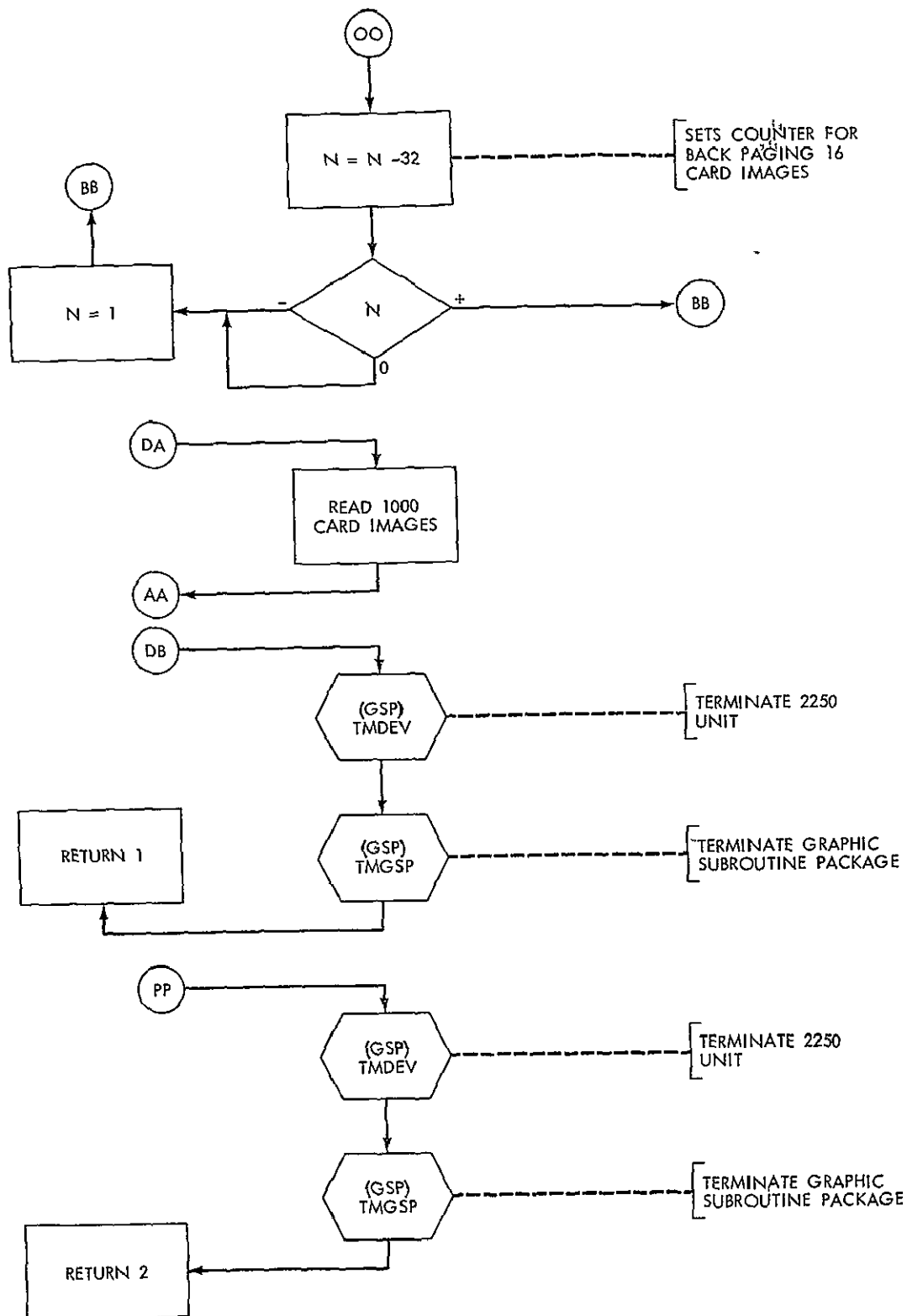
SUBROUTINE DISPLAY (continued)



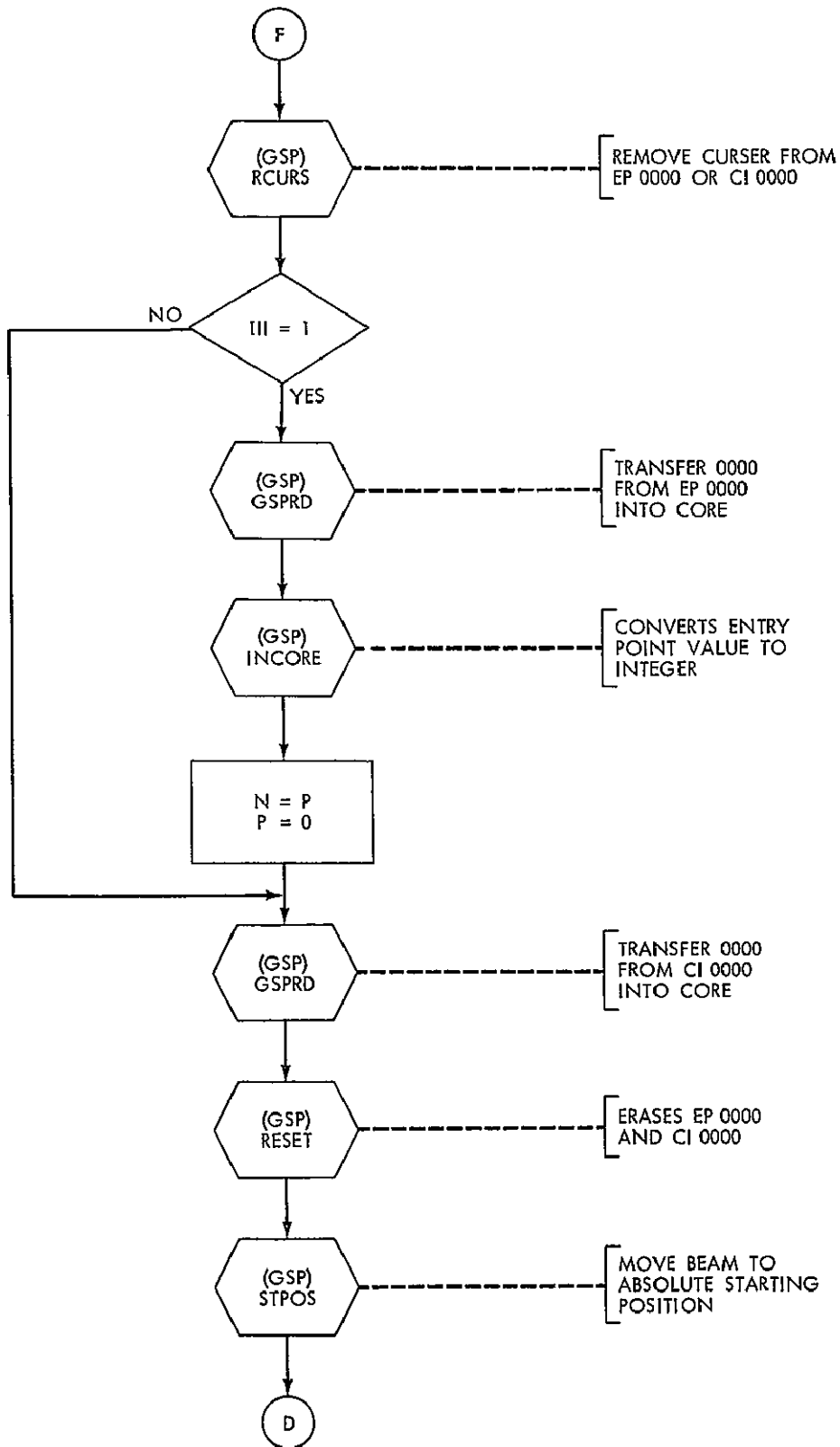
SUBROUTINE DISPLAY (continued)



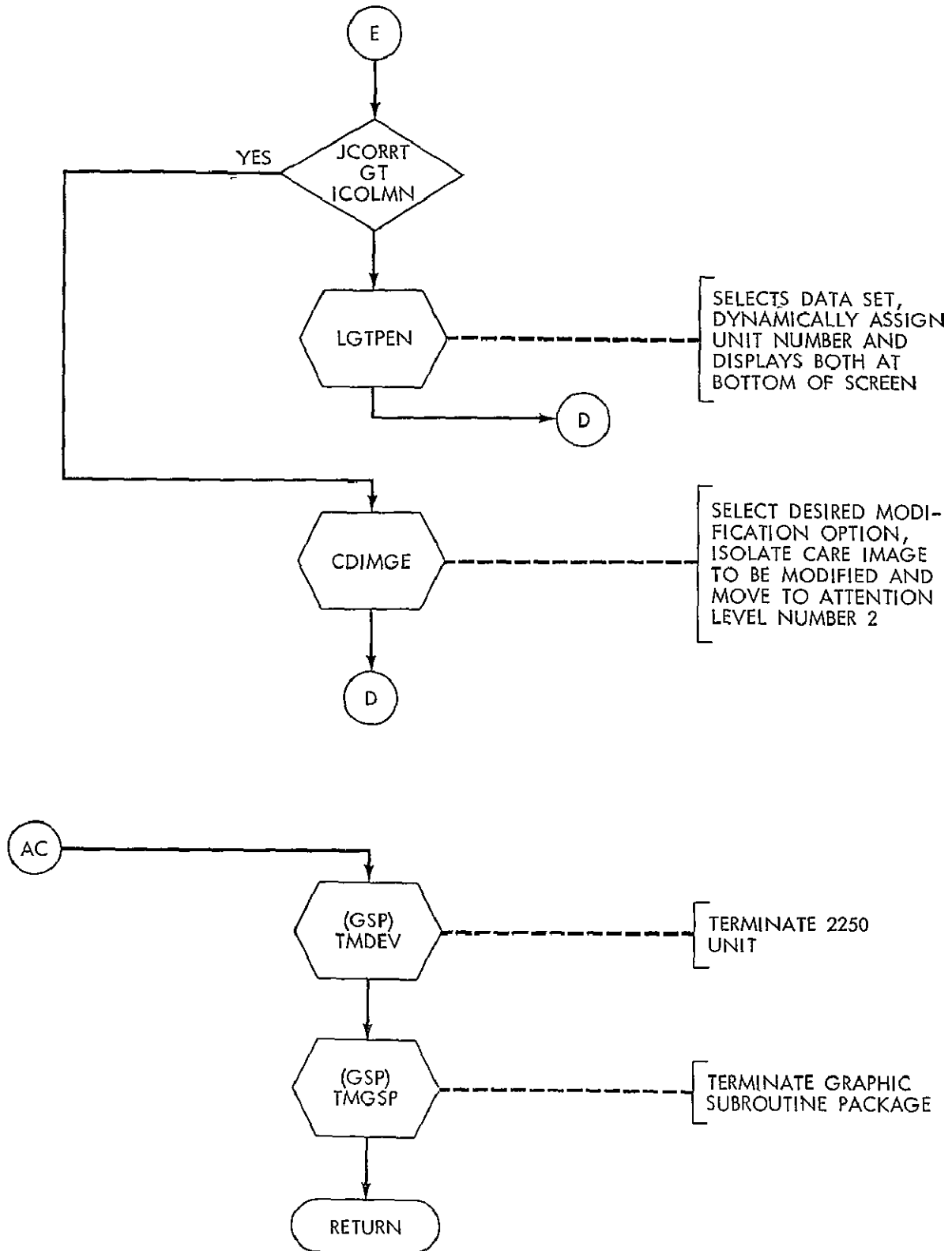
SUBROUTINE DISPLAY (continued)



SUBROUTINE DISPLAY (continued)

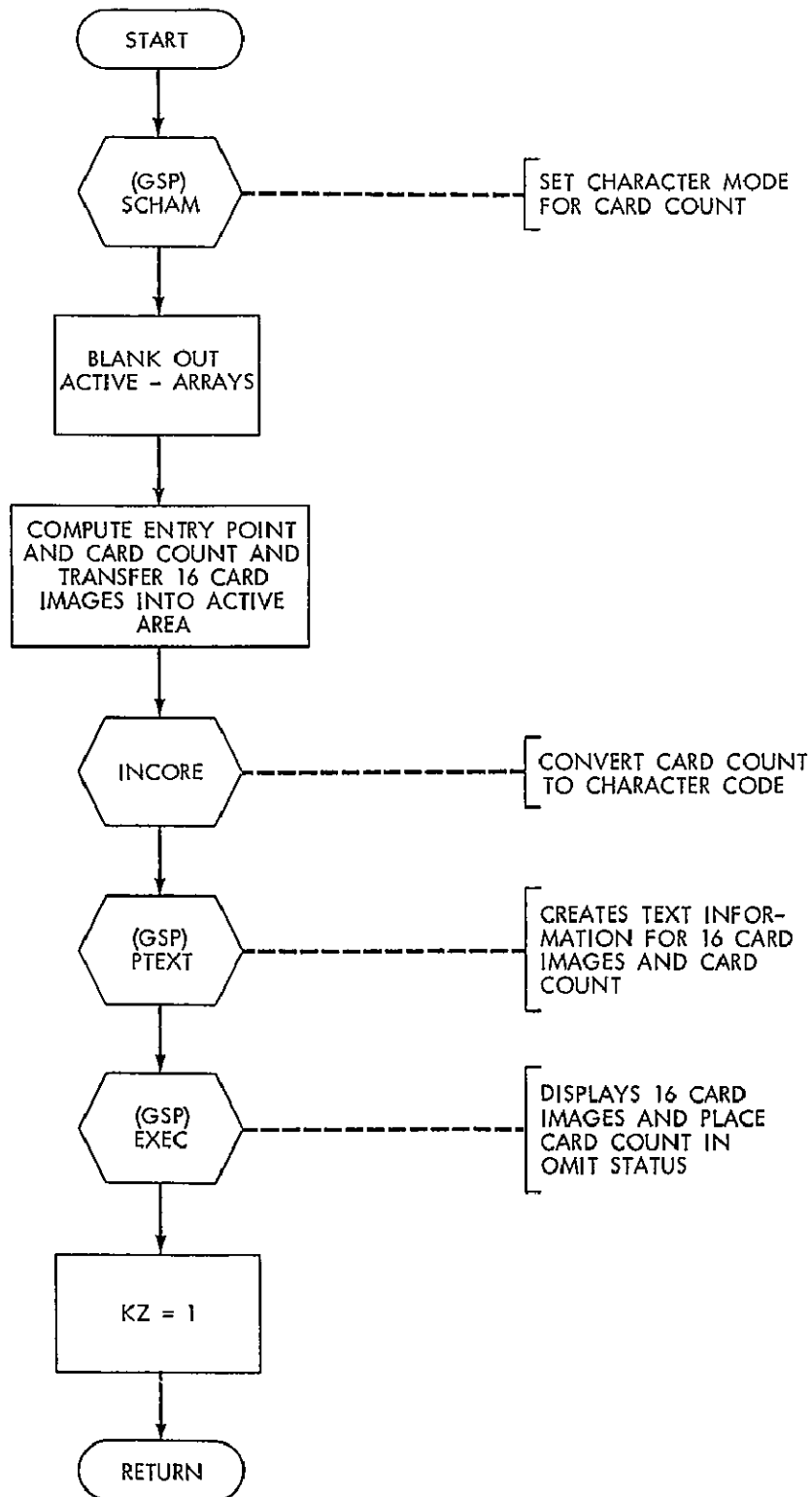


SUBROUTINE DISPLAY (continued)



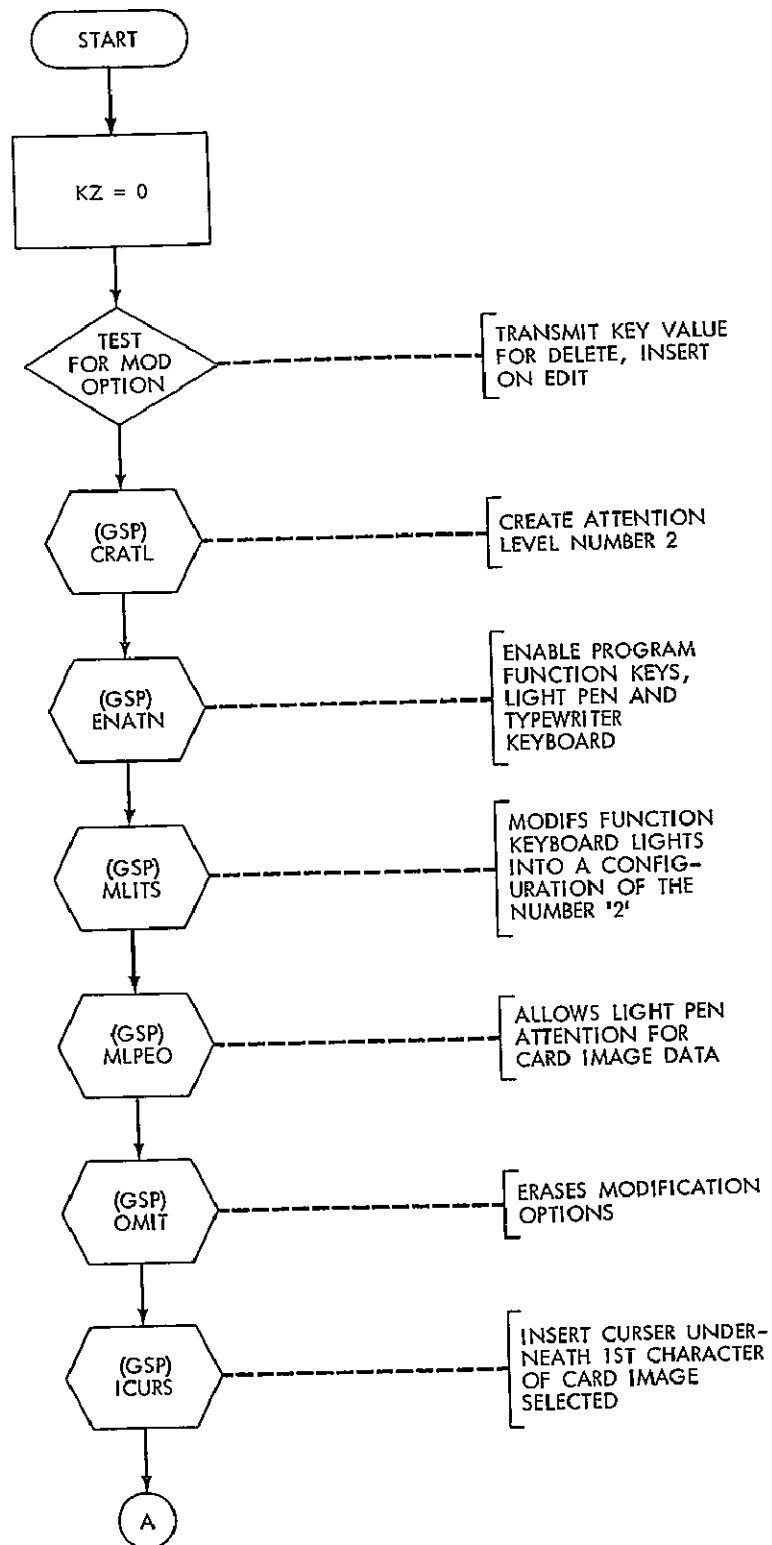
APPENDIX II-B

SUBROUTINE GETCDS

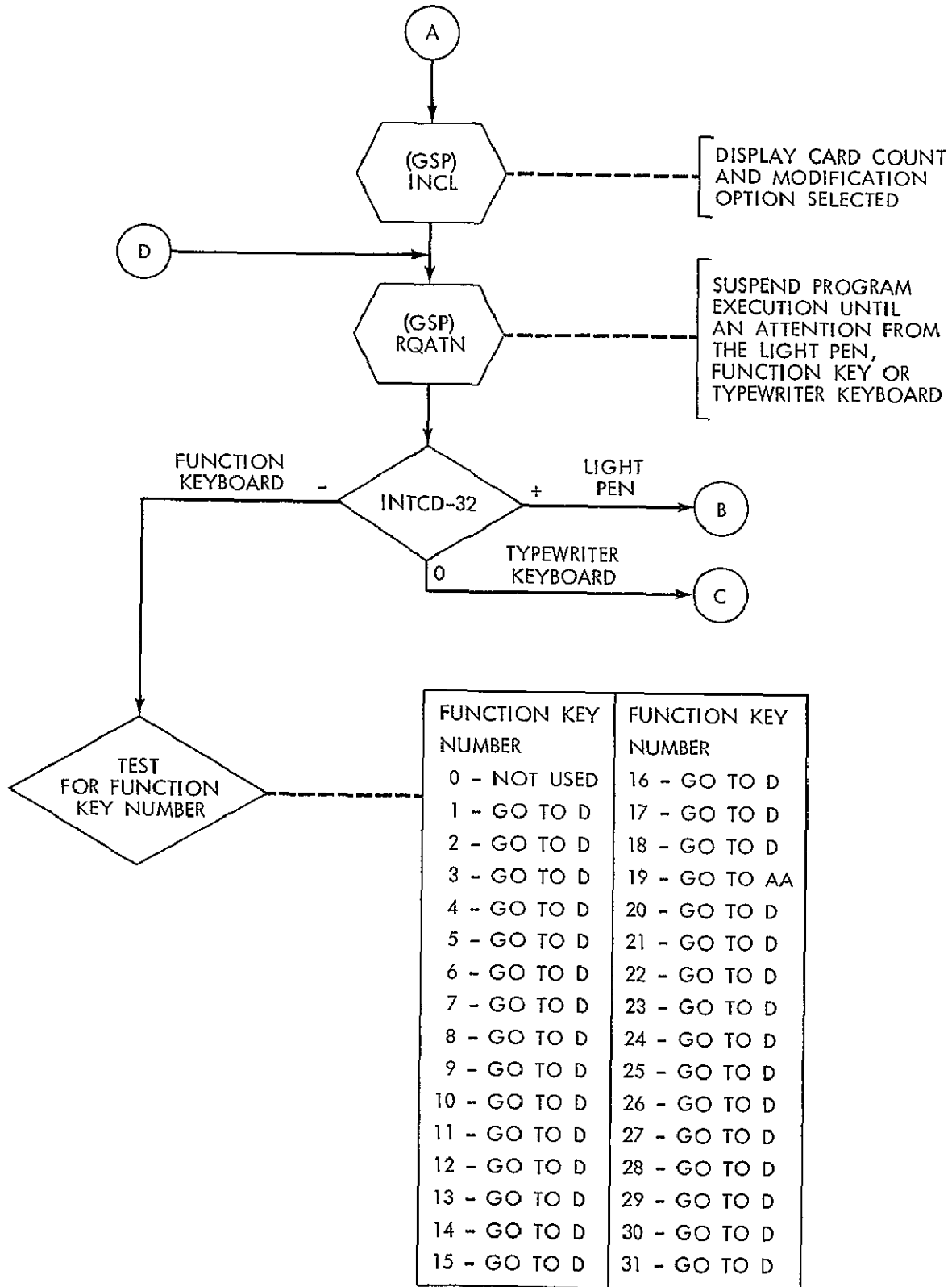


APPENDIX II-C

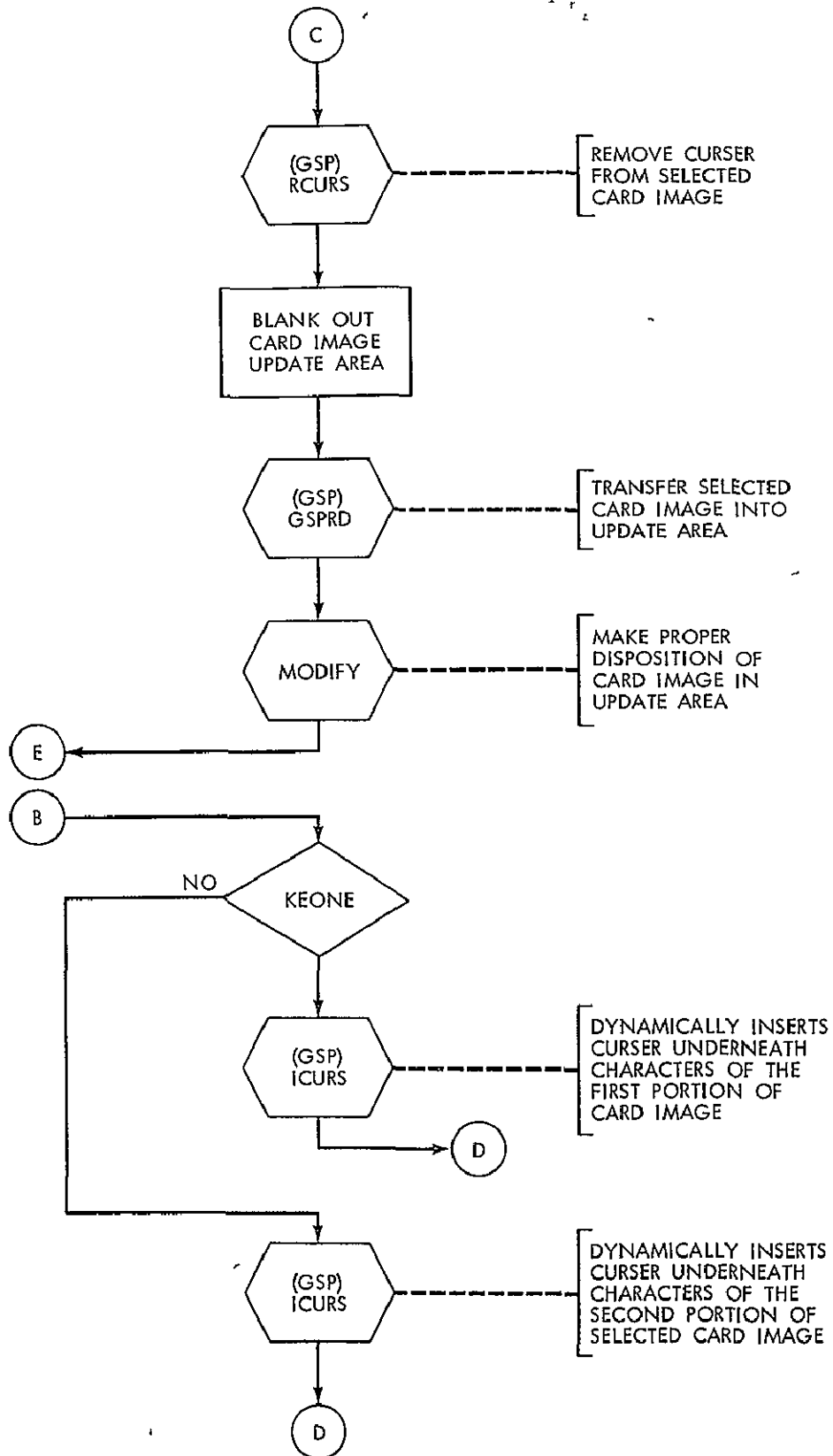
SUBROUTINE CDIMGE



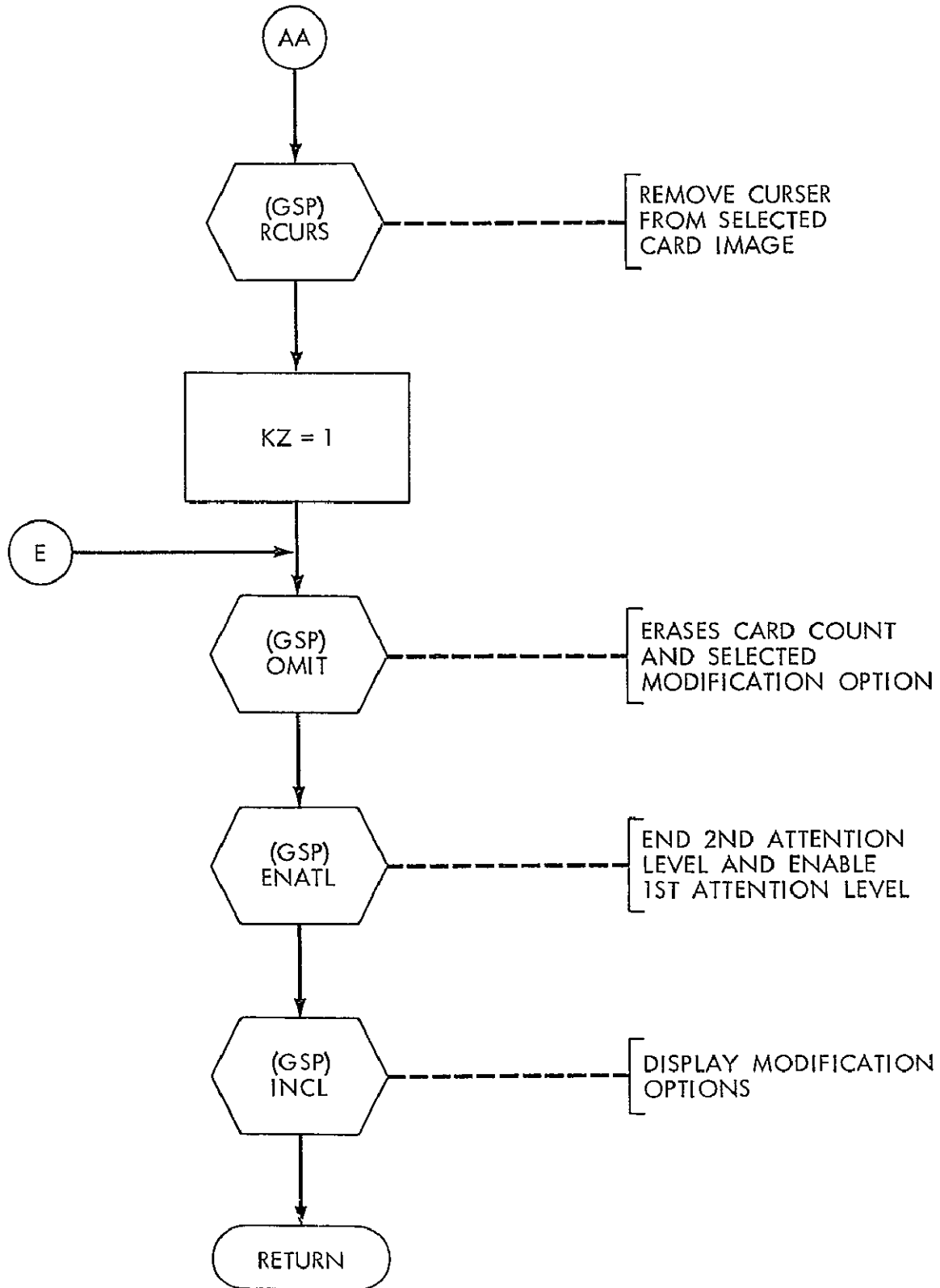
SUBROUTINE CDIMGE (continued)



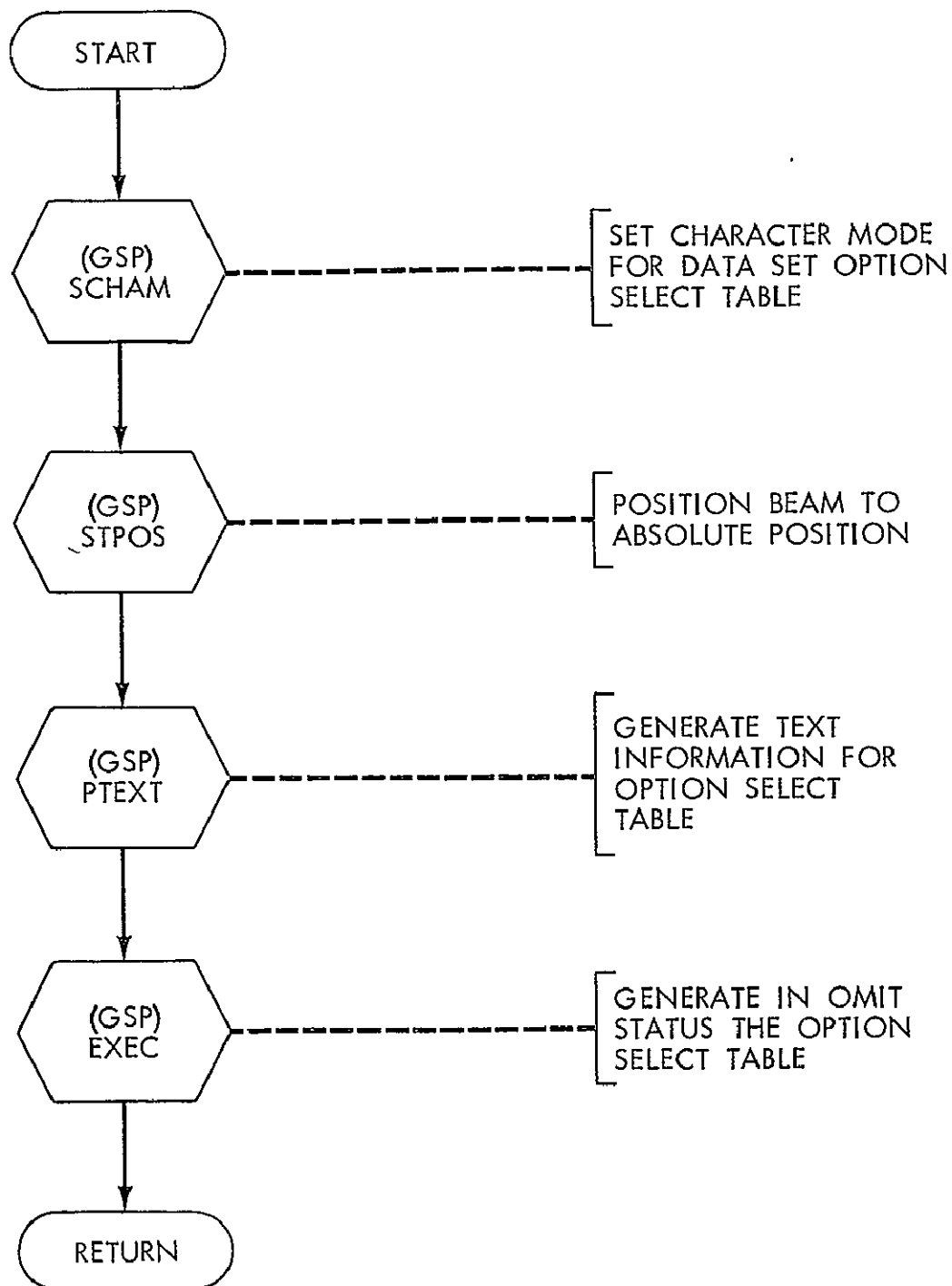
SUBROUTINE CDIMGE (continued)



SUBROUTINE CDIMGE (continued)

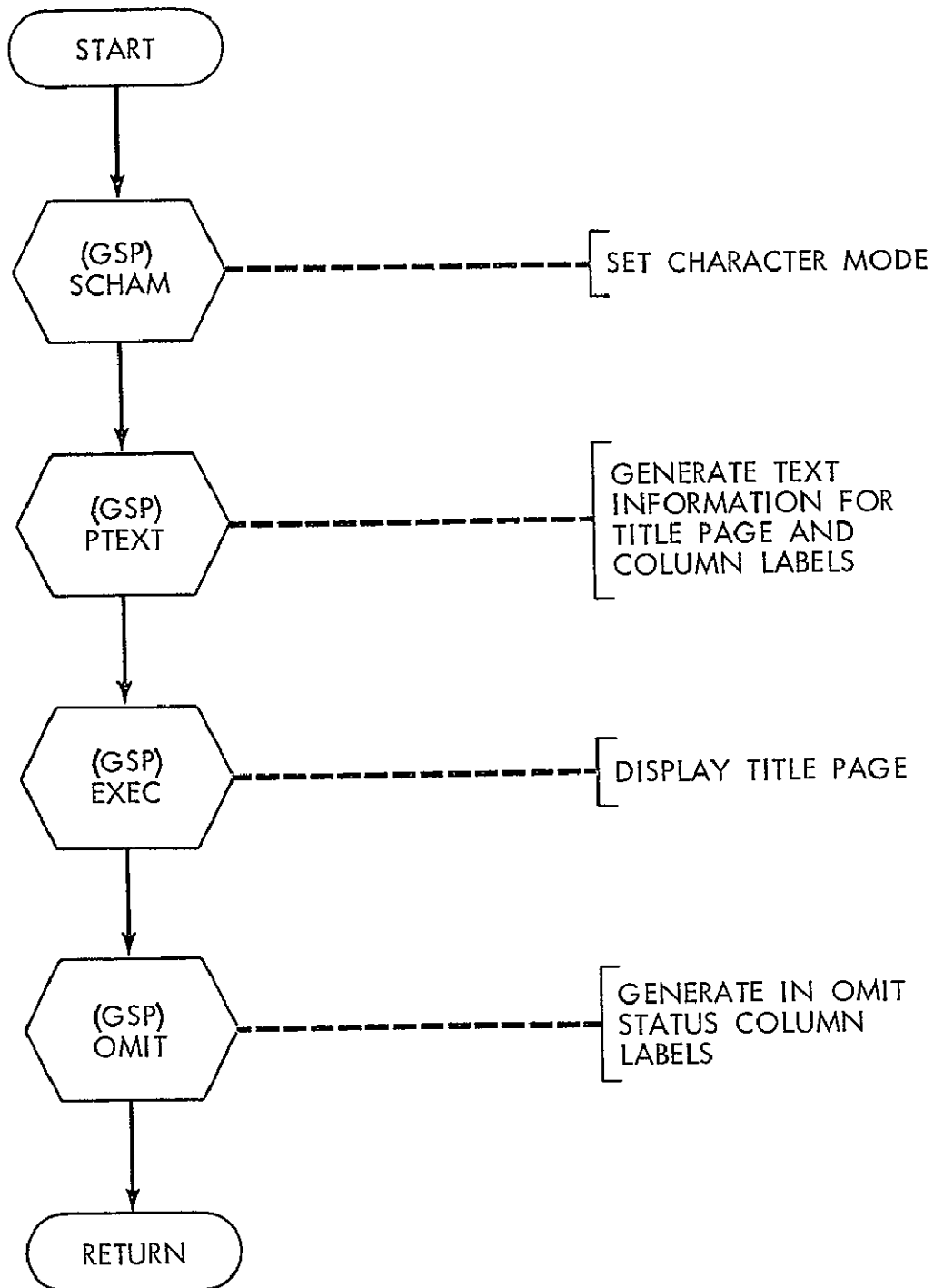


APPENDIX II-D
SUBROUTINE DSTBLE



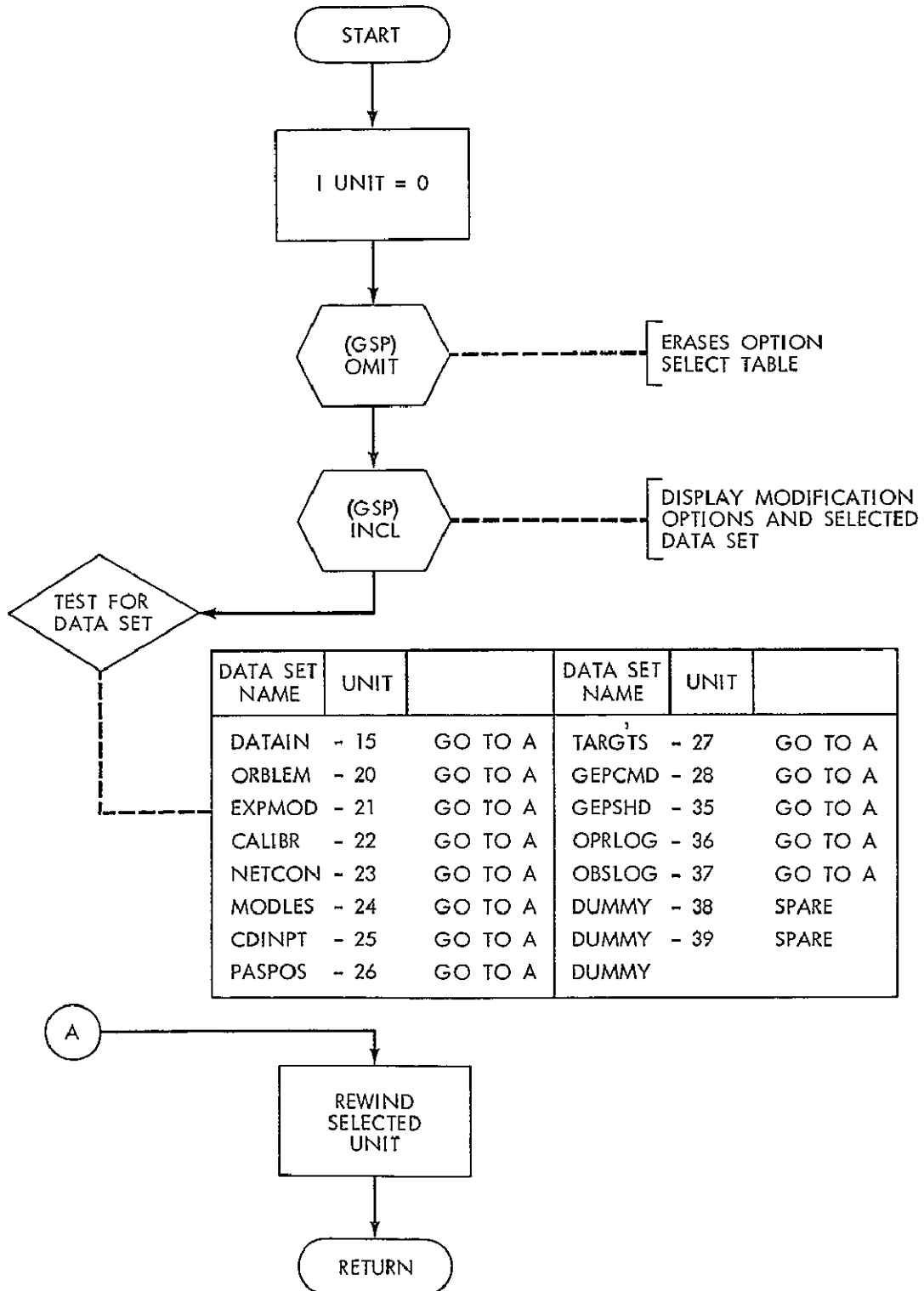
APPENDIX II-E

SUBROUTINE TITLPG

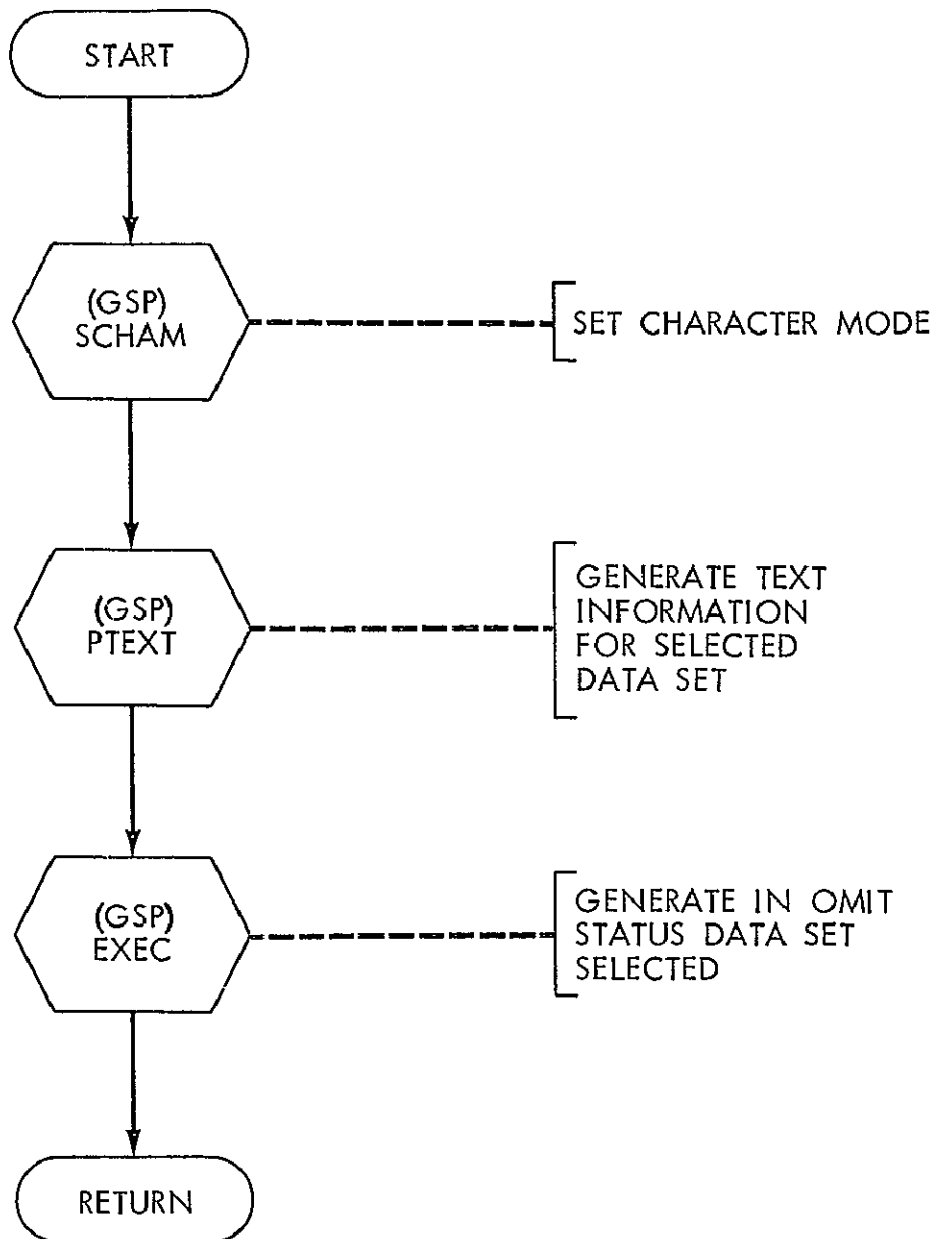


APPENDIX II-F

SUBROUTINE LGTPEN

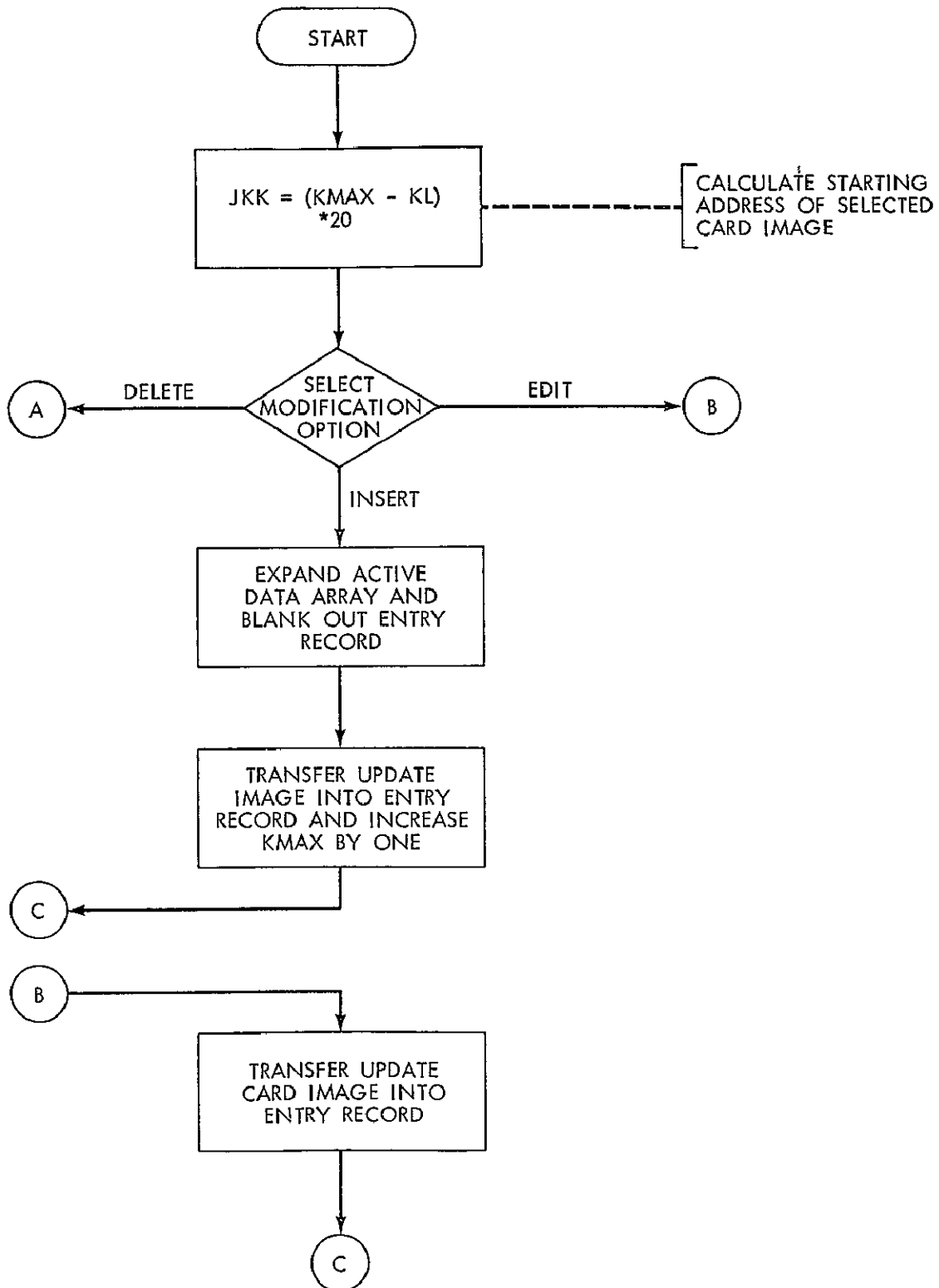


APPENDIX II-G
SUBROUTINE MODSET

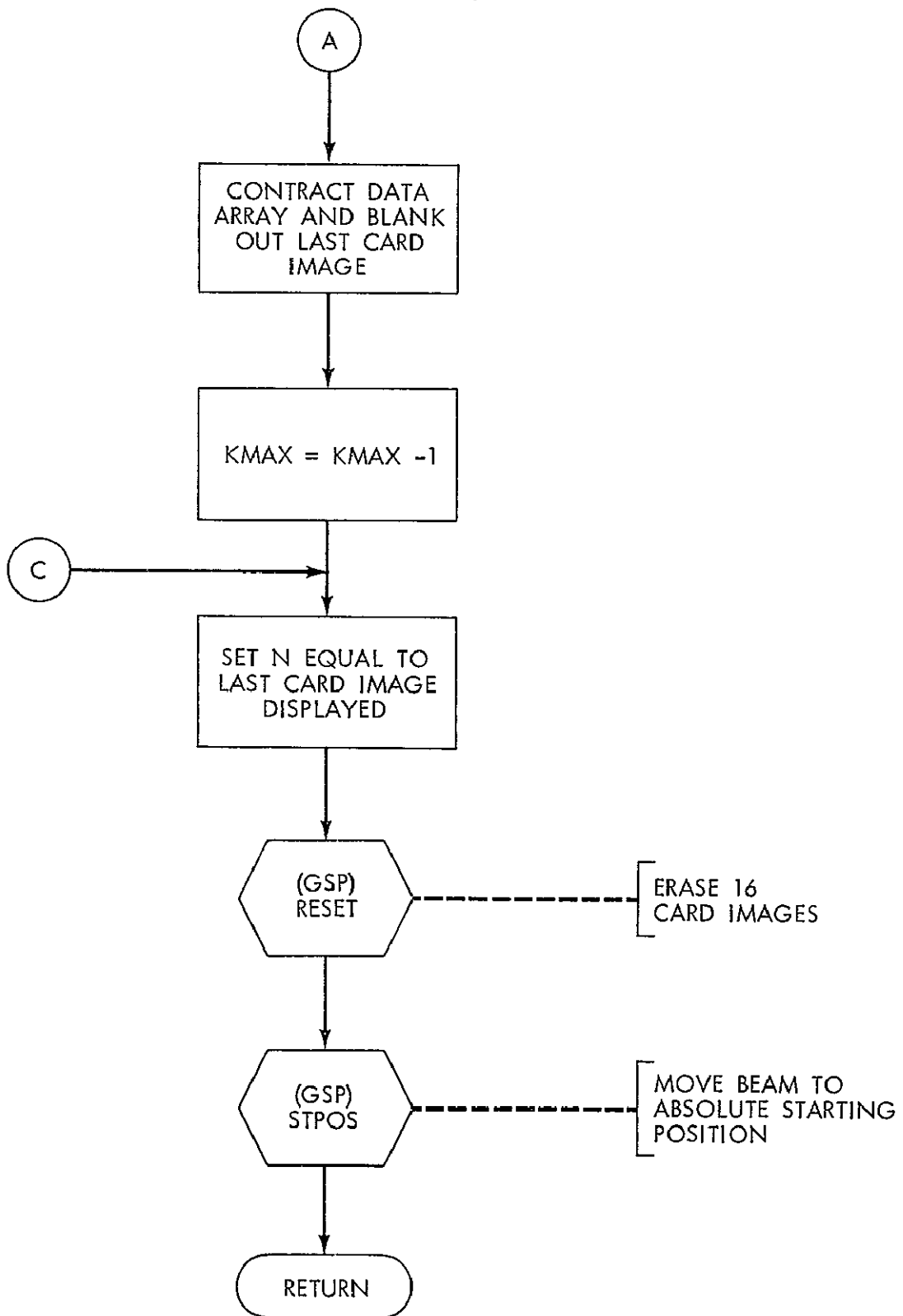


APPENDIX II-H

SUBROUTINE MODIFY



SUBROUTINE MODIFY (continued)



APPENDIX III-A

SUBROUTINE DISPLAY

	SUBROUTINE DISPLAY (IERNUM,*,~)	DISP0010
	COMMON/SAVE/JCORRT	DISP0020
	COMMON/GSET/IGDS1,IGDS2,IGDS3,IGDS4,IGDS5,IGDS6,IGDS7,IGDS8,IGDS9	DISP0030
	COMMON/ILINE/L(16)	DISP0040
	COMMON/JKKARD/JK(32)	DISP0050
	COMMON/ACTARY/DSAREA(20000)	DISP0060
	COMMON/ITLINE/IL(16)	DISP0070
	COMMON/KEEPN/IS(16)	DISP0080
	COMMON/LDCONT/ICCONT(16)	DISP0090
	COMMON/I2250/IGRAFD,IATL,IATL1	DISP0100
	COMMON/NULL/IGNORE(1)	DISP0110
	DIMENSION IN(1),INCRT(1),CSTART(1),NSTART(1),IMSTAT(1),IKSTAT(1)	DISP0120
	DIMENSION INTARY(10),TARY(10)	DISP0130
	DATA V3/' '	DISP0140
	DATA CSTART/'EP',NSTART/'0000',IMSTAT/'CI',IKSTAT/'0000'	DISP0150
	EQUIVALENCE(INTARY(1),TARY(1))	DISP0160
	INTEGER P*4	DISP0170
C	DESIGNATE THAT PROGRAM BE ABNORMALLY TERMINATED AND A DUMP	DISP0180
C	BE PRODUCED	DISP0190
	IGNORE(1)=~5	DISP0200
	KZ=1	DISP0210
	IL(1)=3800	DISP0220
	IL(2)=3500	DISP0230
	IL(3)=3300	DISP0240
	IL(4)=3100	DISP0250
	IL(5)=2800	DISP0260
	IL(6)=2600	DISP0270
	IL(7)=2400	DISP0280
	IL(8)=2100	DISP0290
	IL(9)=1900	DISP0300
	IL(10)=1600	DISP0310
	IL(11)=1300	DISP0320
	IL(12)=1100	DISP0330
	IL(13)=890	DISP0340
	IL(14)=600	DISP0350
	IL(15)=400	DISP0360
	IL(16)=190	DISP0370
	ICOLMN=1600	DISP0380
C	ESTABLISHES COMMUNICATION BETWEEN USER'S PROGRAM AND GSP AND	DISP0390
C	DEFINES THE NULL VARIABLE	DISP0400
	CALL INGSP(IGSP,IGNORE)	DISP0410
C	IDENTIFIES 2250 AS UNIT 49 , IGRAFD IS THE NAME USED TO REFER	DISP0420
C	TO THIS 2250	DISP0430
	CALL INDEV(IGSP,49,IGRAFD)	DISP0440
C	CREATES GRAPHIC DATA SETS AND ASSOCIATES THEM WITH THE 2250	DISP0450
C	IDENTIFIED AS IGRAFD	DISP0460
	CALL INGDS(IGRAFD,IGDS1)	DISP0470
	CALL INGDS(IGRAFD,IGDS2)	DISP0480
	CALL INGDS(IGRAFD,IGDS3)	DISP0490
	CALL INGDS(IGRAFD,IGDS4)	DISP0500
	CALL INGDS(IGRAFD,IGDS5)	DISP0510
	CALL INGDS(IGRAFD,IGDS6)	DISP0520
	CALL INGDS(IGRAFD,IGDS7)	DISP0530
	CALL INGDS(IGRAFD,IGDS8)	DISP0540
	CALL INGDS(IGRAFD,IGDS9)	DISP0550
	CALL INGDS(IGRAFD,IGDS6)	DISP0560
	CALL SGRAM(IGDS1,2)	DISP0570
	CALL SGRAM(IGDS2,2)	DISP0580
	CALL SGRAM(IGDS3,2)	DISP0590
	CALL SGRAM(IGDS4,2)	DISP0600
	CALL SGRAM(IGDS5,2)	DISP0610
	CALL SGRAM(IGDS6,2)	DISP0620
	CALL SGRAM(IGDS7,2)	DISP0630

SUBROUTINE DISPLAY (continued)

	CALL SGRAM(IGDS8,2)	DISP0640
	CALL SGRAM(IGDS9,2)	DISP0650
	CALL SGRAM(KGDS6,2)	DISP0660
C	CREATES AN ATTENTION LEVEL	DISP0670
	LEVFL1=2	DISP0680
	CALL CRATL(IGRAFD,IATL,LEVEL1)	DISP0690
C	ENABLE PROGRAM FUNCTION KEYS , LIGHT PEN AND KEYBOARD	DISP0700
	CALL ENATN(IATL,1,-31,32,34)	DISP0710
	CALL SALRM(IGRAFD)	DISP0720
	CALL MLITS(IATL,4,2,5,7,13,19,23,-27)	DISP0730
C	ALLOW LIGHT PEN ATTENTION FOR DATA SET IGDS1	DISP0740
	CALL SLPAT(IGDS1,1)	DISP0750
	CALL SLPAT(IGDS2,2)	DISP0760
	CALL SLPAT(IGDS3,2)	DISP0770
	CALL SLPAT(IGDS4,1)	DISP0780
	CALL SLPAT(IGDS5,2)	DISP0790
	CALL SLPAT(IGDS6,2)	DISP0800
	CALL SLPAT(IGDS7,1)	DISP0810
	CALL SLPAT(IGDS8,2)	DISP0820
	CALL SLPAT(IGDS9,2)	DISP0830
	CALL SLPAT(KGDS6,2)	DISP0840
	CALL MODSET	DISP0850
	CALL TITLPG	DISP0860
	CALL MLPEQ(IATL,2,4,1)	DISP0870
100	CONTINUE	DISP0880
	IF(KZ.NE.1) GO TO 205	DISP0890
	CALL RQATN(IATL,INTCD,2,INTARY,0,-31,32,34)	DISP0900
	IF(INTCD-32)102,232,234	DISP0910
102	GO TO(201,202,203,204,205,206,207,208,209,210,211,212,213,214,215,216,217,218,219,220,221,222,223,224,225,226,227,228,229,230,231),INTCD	DISP0920
201	CONTINUE	DISP0930
	IJKL=12000	DISP0940
	IF(IUNIT.EQ.26) IJKL=20000	DISP0950
	DO 302 I=1,IJKL	DISP0960
	DSAREA(I)=V3	DISP0970
302	CONTINUE	DISP0980
	READ(IUNIT,80,END=85)(DSAREA(I),I=1,20000)	DISP0990
80	FORMAT(20A4)	DISP1000
85	KMAX=I/20	DISP1010
	N=1	DISP1020
	IF (IUNIT .NE. 26) REWIND IUNIT	DISP1030
	CALL SALRM(IGRAFD)	DISP1040
	CALL SALRM(IGRAFD)	DISP1050
	GO TO 100	DISP1060
202	CALL GETCDS(IUNIT,N,KMAX)	DISP1070
	KZ=1	DISP1080
	GO TO 100	DISP1090
203	CONTINUE	DISP1100
	CALL OMIT(IGDS4)	DISP1110
	CALL OMIT(IGDS9,IGNORE,IS(KDSET))	DISP1120
	CALL INCL(IGDS6)	DISP1130
	CALL INCL(IGDS8)	DISP1140
	CALL INCL(IGDS7)	DISP1150
	IUNIT=0	DISP1160
	N=1	DISP1170
	GO TO 100	DISP1180
204	CONTINUE	DISP1190
	CALL RESET(IGDS5)	DISP1200
	CALL RESET(IGDS3)	DISP1210
	CALL DSTBLE	DISP1220
	GO TO 100	DISP1230
205	CONTINUE	DISP1240
		DISP1250
		DISP1260

SUBROUTINE DISPLAY (continued)

CALL RESET(IGDS1)	DISP1270
CALL STPOS(IGDS1,10.0,3975.0)	DISP1280
IF(K7.NE.1) GO TO 202	DISP1290
GO TO 100	DISP1300
206 CONTINUE	DISP1310
CALL INCL(IGDS2)	DISP1320
GO TO 100	DISP1330
207 CONTINUE	DISP1340
CALL OMIT(IGDS2)	DISP1350
GO TO 100	DISP1360
208 CONTINUE	DISP1370
N=N-24	DISP1380
IF(N) 308,308,202	DISP1390
308 N=1	DISP1400
GO TO 202	DISP1410
209 CONTINUE	DISP1420
CALL SCHAM(KGDS6,3)	DISP1430
III=0	DISP1440
IF(N.NE.1) GO TO 309	DISP1450
CALL PTEXT(KGDS6,CSTART(1),2,IGNORE,JKARD,1,10.0,10.0)	DISP1460
CALL PTEXT(KGDS6,NSTART(1),4,IGNORE,JKARD,1,110.0,10.0)	DISP1470
III=1	DISP1480
309 CONTINUE	DISP1490
CALL PTEXT(KGDS6,IMSTAT(1),2,IGNORE,IIMGE,1,350.0,10.0)	DISP1500
CALL PTEXT(KGDS6,IKSTAT(1),4,IGNORE,IKMGE,1,450.0,10.0)	DISP1510
CALL EXEC(KGDS6)	DISP1520
CALL ICURS(KGDS6,IGNORE,IKMGE,1)	DISP1530
IF(III.NE.1) GO TO 100	DISP1540
CALL RCURS(KGDS6)	DISP1550
CALL ICURS(KGDS6,IGNORE,JKARD,1)	DISP1560
GO TO 100	DISP1570
210 CONTINUE	DISP1580
N=N-16	DISP1590
IF(N.EQ.0) N=1	DISP1600
GO TO 202	DISP1610
211 CONTINUE	DISP1620
CALL INCORE(INCORT,NCORRT,5,1,4,0)	DISP1630
GO TO 550	DISP1640
212 CONTINUE	DISP1650
CALL INCORE(INCORT,NCORRT,5,1,4,0)	DISP1660
GO TO 235	DISP1670
213 CONTINUE	DISP1680
CALL INCORE(INCORT,NCORRT,5,1,4,0)	DISP1690
GO TO 100	DISP1700
214 CONTINUE	DISP1710
REWIND IUNIT	DISP1720
KOUT=KMAX*20	DISP1730
WRITE(IUNIT,80)(DSARF A(I),I=1,KOUT)	DISP1740
END FILE IUNIT	DISP1750
REWIND IUNIT	DISP1760
CALL SALRM(IGRAFD)	DISP1770
IUNIT=0	DISP1780
GO TO 100	DISP1790
215 CONTINUE	DISP1800
CALL RESET(IGDS9)	DISP1810
CALL STPOS(IGDS9,1600.0,10.0)	DISP1820
CALL MODSET	DISP1830
CALL SALRM(IGRAFD)	DISP1840
GO TO 100	DISP1850
216 CONTINUE	DISP1860
N=N-32	DISP1870
IF(N) 316,316,202	DISP1880
316 N=1	DISP1890

SUBROUTINE DISPLAY (continued)

GO TO 202	DISP1900
217 CONTINUE	DISP1910
GO TO 100	DISP1920
218 CONTINUE	DISP1930
GO TO 100	DISP1940
219 CONTINUE	DISP1950
GO TO 100	DISP1960
220 CONTINUE	DISP1970
GO TO 100	DISP1980
221 READ(IUNIT,80,END=85)(DSAREA(I),I=1,20000)	DISP1990
GO TO 201	DISP2000
222 CONTINUE	DISP2010
GO TO 100	DISP2020
223 CONTINUE	DISP2030
GO TO 100	DISP2040
224 CONTINUE	DISP2050
GO TO 100	DISP2060
225 CONTINUE	DISP2070
GO TO 100	DISP2080
226 CONTINUE	DISP2090
GO TO 100	DISP2100
227 CONTINUE	DISP2110
GO TO 100	DISP2120
228 CONTINUE	DISP2130
GO TO 100	DISP2140
230 CONTINUE	DISP2150
CALL TMDEV(IGRAFD)	DISP2160
CALL TMGSP(IGSP)	DISP2170
RETURN 1	DISP2180
231 CONTINUE	DISP2190
CALL TMDEV(IGRAFD)	DISP2200
CALL TMGSP(IGSP)	DISP2210
RETURN 2	DISP2220
232 CONTINUE	DISP2230
CALL RCURS(KGDS6)	DISP2240
IF(III.NE 1) GO TO 332	DISP2250
CALL GSPRD(KGDS6,IN,4,1,KTCODE,IGNORE,JKARD)	DISP2260
CALL INCORE(IN,P,5,1,4,0)	DISP2270
N=P	DISP2280
P=0	DISP2290
332 CALL GSPRD(KGDS6,INCORT,4,1,JTCODE,I(IGNORE,IKNCE)	DISP2300
CALL RESET(KGDS6)	DISP2310
CALL STPOS(KGDS6,10 0,3925 0)	DISP2320
GO TO 100	DISP2330
234 CONTINUE	DISP2340
NCORRT=TARY(9)+ 01	DISP2350
JCORRT=TARY(8)+ 01	DISP2360
IF(JCORRT.GT ICOLMN) GO TO 550	DISP2370
235 CALL LGTPEN(IUNIT,NCORRT,KDSET)	DISP2380
GO TO 100	DISP2390
550 DO 549 I=1,10	DISP2400
KIMGE=I	DISP2410
IF(II(1) LT.NCORRT) GO TO 551	DISP2420
549 CONTINUE	DISP2430
551 CONTINUE	DISP2440
KIMGE1=KIMGE*2-1	DISP2450
KIMGE2=KIMGE*2	DISP2460
KYONE=KIMGE1	DISP2470
KYONE=JK(KIMGE1)	DISP2480
KYTWO=JK(KIMGE2)	DISP2490
KL=ICCONT(KIMGE)	DISP2500
KKEY=L(KIMGE)	DISP2510
CALL CDINGE(IGRAFD,KYONE,KYTWO,KKEY,KL,KMAX,N,K7)	DISP2520
GO TO 100	DISP2530
229 CALL TMDEV(IGRAFD)	DISP2540
CALL TMGSP(IGSP)	DISP2550
RETURN	DISP2560
END	DISP2570

0257 CARDS

APPENDIX III-B

SUBROUTINE GETCDS

SUBROUTINE GETCDS(IUNIT,N,KMAX)		GCD\$0010
DATA V2/' '		GCD\$0020
COMMON/KHAREA/CDAREA(320)		GCD\$0030
COMMON/KBCDS/KCOUNT(16)		GCD\$0040
COMMON/GSET/IGDS1,IGDS2,IGDS3,IGDS4,IGDS5,IGDS6,IGDS7,IGDS8,IGDS9		GCD\$0050
COMMON/ILINE/L(16)		GCD\$0060
COMMON/CDCONT/ICCONT(16)		GCD\$0070
COMMON/JKKARD/JK(32)		GCD\$0080
COMMON/ACTARY/DSAREA(20000)		GCD\$0090
CALL SCHAM(IGDS1,3)		GCD\$0100
DO 45 I=1,16		GCD\$0110
ICCONT(I)=0		GCD\$0120
KCOUNT(I)=0		GCD\$0130
45 CONTINUE		GCD\$0140
DO 84 I=1,320		GCD\$0150
CDAREA(I)=V2		GCD\$0160
84 CONTINUE		GCD\$0170
K=16		GCD\$0180
JJ=N*20-19		GCD\$0190
JJJ=JJ+319		GCD\$0200
IK=0		GCD\$0210
DO 95 I=JJ,JJJ		GCD\$0220
IK=IK+1		GCD\$0230
CDAREA(IK)=DSAREA(I)		GCD\$0240
95 CONTINUE		GCD\$0250
MM=N/16+1		GCD\$0260
KK=MOD(N,16)		GCD\$0270
MMM=MM*16-16+(KK-1)		GCD\$0280
IF (N EQ. 1) MMM=0		GCD\$0290
DO 1111 I=1,K		GCD\$0300
ICCONT(I)=MMM+I		GCD\$0310
CALL INCORE(ICCONT(I),KCOUNT(I),19,1,4,0)		GCD\$0320
1111 CONTINUE		GCD\$0330
N=N+K		GCD\$0340
CALL PTEXT(IGDS1,CDAREA(1),72,IGNORE,JK(1),1,10.0,3925.0)		GCD\$0350
CALL PTEXT(IGDS1,CDAREA(19),8,IGNORE,JK(2),1,10.0,3840.0)		GCD\$0360
CALL PTEXT(IGDS1,CDAREA(21),72,IGNORE,JK(3),1,10.0,3695.0)		GCD\$0370
CALL PTEXT(IGDS1,CDAREA(39),8,IGNORE,JK(4),1,10.0,3610.0)		GCD\$0380
CALL PTEXT(IGDS1,CDAREA(41),72,IGNORE,JK(5),1,10.0,3460.0)		GCD\$0390
CALL PTEXT(IGDS1,CDAREA(59),8,IGNORE,JK(6),1,10.0,3375.0)		GCD\$0400
CALL PTEXT(IGDS1,CDAREA(61),72,IGNORE,JK(7),1,10.0,3225.0)		GCD\$0410
CALL PTEXT(IGDS1,CDAREA(79),8,IGNORE,JK(8),1,10.0,3140.0)		GCD\$0420
CALL PTEXT(IGDS1,CDAREA(81),72,IGNORE,JK(9),1,10.0,2990.0)		GCD\$0430
CALL PTEXT(IGDS1,CDAREA(99),8,IGNORE,JK(10),1,10.0,2905.0)		GCD\$0440
CALL PTEXT(IGDS1,CDAREA(101),72,IGNORE,JK(11),1,10.0,2755.0)		GCD\$0450
CALL PTEXT(IGDS1,CDAREA(119),8,IGNORE,JK(12),1,10.0,2670.0)		GCD\$0460
CALL PTEXT(IGDS1,CDAREA(121),72,IGNORE,JK(13),1,10.0,2520.0)		GCD\$0470
CALL PTEXT(IGDS1,CDAREA(139),8,IGNORE,JK(14),1,10.0,2435.0)		GCD\$0480
CALL PTEXT(IGDS1,CDAREA(141),72,IGNORE,JK(15),1,10.0,2285.0)		GCD\$0490
CALL PTEXT(IGDS1,CDAREA(159),8,IGNORE,JK(16),1,10.0,2200.0)		GCD\$0500
CALL PTEXT(IGDS1,CDAREA(161),72,IGNORE,JK(17),1,10.0,2050.0)		GCD\$0510
CALL PTEXT(IGDS1,CDAREA(179),8,IGNORE,JK(18),1,10.0,1965.0)		GCD\$0520
CALL PTEXT(IGDS1,CDAREA(181),72,IGNORE,JK(19),1,10.0,1815.0)		GCD\$0530
CALL PTEXT(IGDS1,CDAREA(199),8,IGNORE,JK(20),1,10.0,1730.0)		GCD\$0540
CALL PTEXT(IGDS1,CDAREA(201),72,IGNORE,JK(21),1,10.0,1580.0)		GCD\$0550
CALL PTEXT(IGDS1,CDAREA(219),8,IGNORE,JK(22),1,10.0,1495.0)		GCD\$0560
CALL PTEXT(IGDS1,CDAREA(221),72,IGNORE,JK(23),1,10.0,1345.0)		GCD\$0570
CALL PTEXT(IGDS1,CDAREA(239),8,IGNORE,JK(24),1,10.0,1260.0)		GCD\$0580
CALL PTEXT(IGDS1,CDAREA(241),72,IGNORE,JK(25),1,10.0,1110.0)		GCD\$0590
CALL PTEXT(IGDS1,CDAREA(259),8,IGNORE,JK(26),1,10.0,1025.0)		GCD\$0600
CALL PTEXT(IGDS1,CDAREA(261),72,IGNORE,JK(27),1,10.0,875.0)		GCD\$0610
CALL PTEXT(IGDS1,CDAREA(279),8,IGNORE,JK(28),1,10.0,790.0)		GCD\$0620
CALL PTEXT(IGDS1,CDAREA(281),72,IGNORE,JK(29),1,10.0,640.0)		GCD\$0630

SUBROUTINE GETCDS (continued)

CALL PTEXT(IGDS1,CDAREA(299),8,IGNORE,JK(30),1,10.0,555.0)	GCDS0640
CALL PTEXT(IGDS1,CDAREA(301),72,IGNORE,JK(31),1,10.0,405.0)	GCDS0650
CALL PTEXT(IGDS1,CDAREA(319),8,IGNORE,JK(32),1,10.0,320.0)	GCDS0660
CALL EXEC(IGDS1)	GCDS0670
CALL SCHAM(IGDS5,1)	GCDS0680
CALL PTEXT(IGDS5,KCOUNT(1),4,IGNORE,L(1),2,650.0,3820.0)	GCDS0690
CALL PTEXT(IGDS5,KCOUNT(2),4,IGNORE,L(2),2,650.0,3590.0)	GCDS0700
CALL PTEXT(IGDS5,KCOUNT(3),4,IGNORE,L(3),2,650.0,3355.0)	GCDS0710
CALL PTEXT(IGDS5,KCOUNT(4),4,IGNORE,L(4),2,650.0,3120.0)	GCDS0720
CALL PTEXT(IGDS5,KCOUNT(5),4,IGNORE,L(5),2,650.0,2885.0)	GCDS0730
CALL PTEXT(IGDS5,KCOUNT(6),4,IGNORE,L(6),2,650.0,2650.0)	GCDS0740
CALL PTEXT(IGDS5,KCOUNT(7),4,IGNORE,L(7),2,650.0,2415.0)	GCDS0750
CALL PTEXT(IGDS5,KCOUNT(8),4,IGNORE,L(8),2,650.0,2180.0)	GCDS0760
CALL PTEXT(IGDS5,KCOUNT(9),4,IGNORE,L(9),2,650.0,1945.0)	GCDS0770
CALL PTEXT(IGDS5,KCOUNT(10),4,IGNORE,L(10),2,650.0,1710.0)	GCDS0780
CALL PTEXT(IGDS5,KCOUNT(11),4,IGNORE,L(11),2,650.0,1475.0)	GCDS0790
CALL PTEXT(IGDS5,KCOUNT(12),4,IGNORE,L(12),2,650.0,1240.0)	GCDS0800
CALL PTEXT(IGDS5,KCOUNT(13),4,IGNORE,L(13),2,650.0,1005.0)	GCDS0810
CALL PTEXT(IGDS5,KCOUNT(14),4,IGNORE,L(14),2,650.0,770.0)	GCDS0820
CALL PTEXT(IGDS5,KCOUNT(15),4,IGNORE,L(15),2,650.0,535.0)	GCDS0830
CALL PTEXT(IGDS5,KCOUNT(16),4,IGNORE,L(16),2,650.0,305.0)	GCDS0840
CALL EXEC(IGDS5)	GCDS0850
RETURN	GCDS0860
END	GCDS0870

0087 CARDS

SUBROUTINE CDMGE

53

2

TABLE

INDEX CAPS

APPENDIX III-E

SUBROUTINE TITLPG

```

SUBROUTINE TITLPG
COMMON/GSFT/ICDS1,IGDS2,IGDS3,ICDS4,IGDS5,IGDS6,IGDS7,ICDS8,ICDS9
COMMON/NULL/ICNORE(1)
DIMENSION CLMDSN(20)
DIMENSION MODSEQ(2)
DIMENSION TLINE1(4),TLINE2(16),TLINE3(14),TLINE4(9),TLINE5(10),
1TLINE6(13)
DATA CLMDSN/'123456789X123456789/123456789312345678941234567895123
14567896123456789/1234567898'/
DATA MODSEQ/'D I E'/
DATA TLINE1/'PROGRAM DISPLAY'/
DATA TLINE2/'PURPOSE____TO DISPLAY AND UPDATE DATA SETS ASSOCIATED
1WITH THE'/
DATA TLINE3/'GODDARD EXPERIMENT PACKAGE OPERATION SCHEDULING SYSTEM
1M'/
DATA TLINE4/'PROGRAMMER____CLARENCE (ROOVER)/
DATA TLINE5/'THEORETICAL ASTROPHYSICS BRANCH,CODE 613'/
DATA TLINE6/'***TO START PROGRAM DEPRESS FUNCTION KEY NUMBER 4'/
SFT CHARACTER MODE
CALL SCHAM(IGDS3,2)
CALL SCHAM(ICDS5,1)
CALL PTEXT(IGDS3,TLINE1(1),15,IGNORE,ITL1,1,1450 0,3800 0)
CALL PTEXT(IGDS5,TLINE2(1),61,IGNORE,ITL2,1,400 0,2900 0)
CALL PTEXT(IGDS5,TLINE3(1),56,IGNORE,ITL3,1,950 0,2700 0)
CALL PTEXT(IGDS5,TLINE4(1),32,IGNORE,ITL4,1,400 0,1800 0)
CALL PTEXT(IGDS5,TLINE5(1),40,IGNORE,ITL5,1,1350 0,1600 0)
CALL PTEXT(IGDS5,TLINE6(1),52,IGNORE,ITL6,1,400 0,1000 0)
CALL EXEC(IGDS3)
CALL EXEC(IGDS5)
CALL SCHAM(IGDS4,2)
CALL PTEXT(IGDS4,MODSEQ(1),8,IGNORE,KAY1,1,3650 0,3820 0)
CALL PTEXT(IGDS4,MODSEQ(1),8,IGNORE,KAY2,1,3650 0,3590 0)
CALL PTEXT(IGDS4,MODSEQ(1),8,IGNORE,KAY3,1,3650 0,3355 0)
CALL PTEXT(IGDS4,MODSEQ(1),8,IGNORE,KAY4,1,3650 0,3120 0)
CALL PTEXT(IGDS4,MODSEQ(1),8,IGNORE,KAY5,1,3650 0,2885 0)
CALL PTEXT(IGDS4,MODSEQ(1),8,IGNORE,KAY6,1,3650 0,2650 0)
CALL PTEXT(IGDS4,MODSEQ(1),8,IGNORE,KAY7,1,3650 0,2415 0)
CALL PTEXT(IGDS4,MODSEQ(1),8,IGNORE,KAY8,1,3650 0,2180 0)
CALL PTEXT(IGDS4,MODSEQ(1),8,IGNORE,KAY9,1,3650 0,1945 0)
CALL PTEXT(IGDS4,MODSEQ(1),8,IGNORE,KAY10,1,3650 0,1710 0)
CALL PTEXT(IGDS4,MODSEQ(1),8,IGNORE,KAY11,1,3650 0,1475 0)
CALL PTEXT(IGDS4,MODSEQ(1),8,IGNORE,KAY12,1,3650 0,1240 0)
CALL PTEXT(IGDS4,MODSEQ(1),8,IGNORE,KAY13,1,3650 0,1005 0)
CALL PTEXT(IGDS4,MODSEQ(1),8,IGNORE,KAY14,1,3650 0,770 0)
CALL PTEXT(IGDS4,MODSEQ(1),8,IGNORE,KAY15,1,3650 0,535 0)
CALL PTEXT(IGDS4,MODSEQ(1),8,IGNORE,KAY16,1,3650 0,305 0)
CALL EXEC(IGDS4)
CALL OMIT(IGDS4)
CALL SCHAM(IGDS2,1)
CALL PTEXT(IGDS2,CLMDSN(1),72,IGNORE,KEY2,1,10 0,4046 0)
CALL PTEXT(IGDS2,CLMDSN(1),72,IGNORE,KEY4,1,10 0,215 0)
CALL EXEC(ICDS2)
CALL OMIT(IGDS2)
RETURN
END

```

0055 CARDS

APPENDIX III-F

SUBROUTINE LGTPEN

```

SUBROUTINE LGTPEN(IUNIT,NCURRT,KDSET)
COMMON/CSET/IGDS1,IGDS2,IGDS3,IGDS4,IGDS5,IGDS6,IGDS7,IGDS8,IGDS9
COMMON/NUIL/IGNORE(1)
COMMON/COMPEN/IL(16)
COMMON/KEEPN/IS(16)
IUNIT=0
IL(1)=3390
NL(2)=3190
NL(3)=2990
NL(4)=2790
NL(5)=2590
NL(6)=2390
NL(7)=2190
NL(8)=1990
NL(9)=1780
NL(10)=1590
NL(11)=1390
NL(12)=1190
NL(13)=990
NL(14)=790
NL(15)=590
NL(16)=390
CALL OMIT(IGDS6)
CALL OMIT(IGDS7)
CALL OMIT(IGDS8)
CALL INCI(IGDS4)
DO 499 I=1,16
  KDSFI=1
  IF(NL(I) LT NCURRT) GO TO 498
499 CONTINUE
498 CONTINUE
  CALL INCL(IGDS9,IGDS9,IS(KDSET))
  GO TO(501,502,503,504,505,506,507,508,509,510,511,512,513,514,515,
    1516),KDSET
501 IUNIT=15
  GO TO 520
502 IUNIT=20
  GO TO 520
503 IUNIT=21
  GO TO 520
504 IUNIT=22
  GO TO 520
505 IUNIT=23
  GO TO 520
506 IUNIT=24
  GO TO 520
507 IUNIT=25
  GO TO 520
508 IUNIT=26
  GO TO 520
509 IUNIT=27
  GO TO 520
510 IUNIT=28
  GO TO 520
511 IUNIT=35
  GO TO 520
512 IUNIT=36
  GO TO 520
513 IUNIT=37
  GO TO 520
514 IUNIT=38
  GO TO 520
515 IUNIT=39

520 REWIND IUNIT
  RETURN
516 IUNIT=5
  RETURN
  END

```

```

TPN00100
TPN00110
TPN00120
TPN00130
TPN00140
TPN00150
TPN00160
TPN00170
TPN00180
TPN00190
TPN00200
TPN00210
TPN00220
TPN00230
TPN00240
TPN00250
TPN00260
TPN00270
TPN00280
TPN00290
TPN00300
TPN00310
TPN00320
TPN00330
TPN00340
TPN00350
TPN00360
TPN00370
TPN00380
TPN00390
TPN00400
TPN00410
TPN00420
TPN00430
TPN00440
TPN00450
TPN00460
TPN00470
TPN00480
TPN00490
TPN00500
TPN00510
TPN00520
TPN00530
TPN00540
TPN00550
TPN00560
TPN00570
TPN00580
TPN00590
TPN00600
TPN00610
TPN00620
TPN00630
TPN00640
TPN00650
TPN00660
TPN00670
TPN00680
TPN00690
TPN00700
TPN00710
TPN00720

```

```

TPN00730
TPN00740
TPN00750
TPN00760
TPN00770

```

CONT CONT

APPENDIX III-G

SUBROUTINE MODSET

SUBROUTINE MODSET	MDST0010
COMMON/KEEPN/IS(16)	MDST0020
COMMON/NKSET/ISETC,ISETD,ISETL	MDST0030
COMMON/NULL/IGNORE(1)	MDST0040
COMMON/GSET/IGDS1,IGDS2,IGDS3,IGDS4,IGDS5,IGDS6,IGDS7,IGDS8,IGDS9	MDST0050
DIMENSION MODIF1(1),MODIF2(2),MODIF3(2)	MDST0060
DIMENSION DSET1(7),DSET2(7),DSET3(7),DSET4(7),DSET5(7),DSET6(7),	MDST0070
DSET7(7),DSET8(7),DSET9(7),DSET10(7),DSET11(7),DSET12(7),	MDST0080
DSET13(7),DSET14(7),DSET15(7),DSET16(7)	MDST0090
DATA MODIF1/'EDIT'/	MDST0100
DATA MODIF2/'INSERT'/	MDST0110
DATA MODIF3/'DELETE'/	MDST0120
DATA DSET1/'DS_NAME DATAIN UNIT# 15'/	MDST0130
DATA DSET2/'DS_NAME ORKELM UNIT# 20'/	MDST0140
DATA DSET3/'DS_NAME EXPMOD UNIT# 21'/	MDST0150
DATA DSET4/'DS_NAME CALIBR UNIT# 22'/	MDST0160
DATA DSET5/'DS_NAME NETCON UNIT# 23'/	MDST0170
DATA DSET6/'DS_NAME MODELS UNIT# 24'/	MDST0180
DATA DSET7/'DS_NAME CDINPT UNIT# 25'/	MDST0190
DATA DSET8/'DS_NAME PASPOS UNIT# 26'/	MDST0200
DATA DSET9/'DS_NAME TARGETS UNIT# 27'/	MDST0210
DATA DSET10/'DS_NAME GEPCMD UNIT# 28'/	MDST0220
DATA DSET11/'DS_NAME GEPSHD UNIT# 35'/	MDST0230
DATA DSET12/'DS_NAME DPRLOG UNIT# 36'/	MDST0240
DATA DSET13/'DS_NAME OBSLOG UNIT# 37'/	MDST0250
DATA DSET14/'DS_NAME CNTMSG UNIT# 37'/	MDST0260
DATA DSET15/'DS_NAME INTPRT UNIT# 39'/	MDST0270
DATA DSET16/'DS_NAME CDREDR UNIT# 5'/	MDST0280
CALL SCHAM(IGDS9,1)	MDST0290
CALL PTEXT(IGDS9,DSET1(1),27,IGNORE,IS(1),2,1600 0,10.0)	MDST0300
CALL PTEXT(IGDS9,DSET2(1),27,IGNORE,IS(2),2,1600 0,10.0)	MDST0310
CALL PTEXT(IGDS9,DSET3(1),27,IGNORE,IS(3),2,1600.0,10.0)	MDST0320
CALL PTEXT(IGDS9,DSET4(1),27,IGNORE,IS(4),2,1600.0,10.0)	MDST0330
CALL PTEXT(IGDS9,DSET5(1),27,IGNORE,IS(5),2,1600.0,10.0)	MDST0340
CALL PTEXT(IGDS9,DSET6(1),27,IGNORE,IS(6),2,1600 0,10.0)	MDST0350
CALL PTEXT(IGDS9,DSET7(1),27,IGNORE,IS(7),2,1600 0,10.0)	MDST0360
CALL PTEXT(IGDS9,DSET8(1),27,IGNORE,IS(8),2,1600 0,10.0)	MDST0370
CALL PTEXT(IGDS9,DSET9(1),27,IGNORE,IS(9),2,1600 0,10.0)	MDST0380
CALL PTEXT(IGDS9,DSET10(1),27,IGNORE,IS(10),2,1600 0,10.0)	MDST0390
CALL PTEXT(IGDS9,DSET11(1),27,IGNORE,IS(11),2,1600 0,10.0)	MDST0400
CALL PTEXT(IGDS9,DSET12(1),27,IGNORE,IS(12),2,1600 0,10.0)	MDST0410
CALL PTEXT(IGDS9,DSET13(1),27,IGNORE,IS(13),2,1600.0,10.0)	MDST0420
CALL PTEXT(IGDS9,DSET14(1),27,IGNORE,IS(14),2,1600 0,10.0)	MDST0430
CALL PTEXT(IGDS9,DSET15(1),27,IGNORE,IS(15),2,1600.0,10.0)	MDST0440
CALL PTEXT(IGDS9,DSET16(1),27,IGNORE,IS(16),2,1600 0,10.0)	MDST0450
CALL PTEXT(IGDS9,MODIF1(1),4,IGNORE,ISETC,2,3600 0,100.0)	MDST0460
CALL PTEXT(IGDS9,MODIF2(1),6,IGNORE,ISETD,2,3600.0,100.0)	MDST0470
CALL PTEXT(IGDS9,MODIF3(1),6,IGNORE,ISETL,2,3600 0,100.0)	MDST0480
CALL EXEC(IGDS9)	MDST0490
RETURN	MDST0500
END	MDST0510

0051 (ARDS

APPENDIX III-H

SUBROUTINE MODIFY

SUBROUTINE MODIFY(KL,KMAX,N)	MDFY0010
COMMON/GSET/IGDS1,IGDS2,IGDS3,IGDS4,IGDS5,IGDS6,IGDS7,IGDS8,IGDS9	MDFY0020
COMMON/KEYBD/UPDATE(20)	MDFY0030
COMMON/NULL/IGNORE(1)	MDFY0040
COMMON/SAVE/JCURRT	MDFY0050
COMMON/CDCONT/ICCONT(16)	MDFY0060
COMMON/ACTARY/DSAREA(20000)	MDFY0070
DATA V4/' '	MDFY0080
KLINE1=3600	MDFY0090
KLINE2=3775	MDFY0100
KLINE3=3900	MDFY0110
JKK=(KMAX-KL)*20	MDFY0120
IF(KLINE3.LT.JCURRT) GO TO 80	MDFY0130
IF(KLINE2.LT.JCURRT) GO TO 50	MDFY0140
IF(KLINE1.LT.JCURRT) GO TO 30	MDFY0150
30 J=KL*20+1	MDFY0160
DO 40 I=1,JKK	MDFY0170
IJ=J-20	MDFY0180
DSAREA(IJ)=DSAREA(J)	MDFY0190
J=J+1	MDFY0200
40 CONTINUE	MDFY0210
JKL=KMAX*20-19	MDFY0220
DO 35 I=1,20	MDFY0230
DSAREA(JKL)=V4	MDFY0240
JKL=JKL+1	MDFY0250
35 CONTINUE	MDFY0260
KMAX=KMAX-1	MDFY0270
GO TO 100	MDFY0280
50 J=KMAX*20	MDFY0290
DO 60 I=1,JKK	MDFY0300
IJ=J+20	MDFY0310
DSAREA(IJ)=DSAREA(J)	MDFY0320
J=J-1	MDFY0330
60 CONTINUE	MDFY0340
KP=KL*20+1	MDFY0350
DO 70 I=1,20	MDFY0360
DSAREA(KP)=UPDATE(I)	MDFY0370
KP=KP+1	MDFY0380
70 CONTINUE	MDFY0390
KMAX=KMAX+1	MDFY0400
GO TO 100	MDFY0410
80 J=KL*20-19	MDFY0420
DO 90 I=1,20	MDFY0430
DSAREA(J)=UPDATE(I)	MDFY0440
J=J+1	MDFY0450
90 CONTINUE	MDFY0460
100 N=ICCONT(1)	MDFY0470
CALL RESET(IGDS1)	MDFY0480
CALL STPOS(IGDS1,10.0,3925.0)	MDFY0490
RETURN	MDFY0500
END	MDFY0510

0051 CARDS